UNIVERSIDADE DE LISBOA
FACULDADE DE CIÊNCIAS
DEPARTAMENTO DE INFORMÁTICA

MOBILE COLLABORATIVE CLOUDLESS COMPUTING

DOUTORAMENTO EM INFORMÁTICA

(Engenharia Informática)

Nuno Miguel Machado Cruz

Tese orientada por: Prof. Doutor Hugo Alexandre Miranda

2015

Documento especialmente elaborado para a obtenção do grau de doutor.

Resumo Estendido

Apesar do crescente poder computacional que tem vindo a caracterizar as novas gerações de *smarphones*, ainda existem aplicações que têm necessidade de delegar a computação num conjunto de servidores existentes na Internet. Este modelo assume uma ligação à Internet sempre disponível e introduz uma latência não negligenciável. A computação em nuvem é um paradigma inovador em que os recursos de um conjunto de servidores são partilhados de forma transparente por um conjunto de utilizadores. Este paradigma, simplifica a gestão dos recursos, criando o conceito de computação elástica, em que cada utilizador pode aumentar ou diminuir facilmente os recursos que lhe estão atribuídos. A tese estuda as vantagens e desafios colocados à aplicação de uma variação do conceito de tradicional computação em nuvem, onde os dispositivos móveis dispensam a conectividade à Internet, definindo uma arquitetura para aplicações móveis com elevado poder computacional, beneficiando da computação móvel colaborativa (CMC). Neste cenário, o poder computacional é obtido a partir de um conjunto de dispositivos móveis, que se coordenam entre si para atingir um objetivo comum - a execução de um conjunto de tarefas propostas por um ou mais participantes. Espera-se que a utilização da CMC permita por um lado obter resultados num tempo inferior ao que leva a delegação da tarefa na nuvem comercial e por outro atenue o impacto das situações em que o acesso à Internet não se encontra disponível e ao mesmo tempo alivie a largura de banda consumida por estas aplicações nos canais de comunicação celular.

A implementação da arquitetura proposta apresenta uma série de desafios à sua implementação. Entre estes incluem-se a privacidade dos dados dos utilizadores, com-

portamentos desviantes que explorem a natureza da CMC, eleição dos fornecedores do poder computacional, cálculo da complexidade das tarefas e impacto no consumo de energia. Os desafios são pela proposta da criação de um *middleware* a que demos o nome de Ambiente de Computação Móvel Colaborativo (ACMC).

Para que a CMC venha a ser adotada pelos utilizadores, o ACMC terá de fornecer mecanismos que permitam estabelecer a confiança entre os dispositivos e incentivar a partilha de recursos. A tarefa de permitir que dispositivos vizinhos executem tarefas localmente torna-os vulneráveis a ataques à confidencialidade e privacidade de dados, e de exaustão de recursos. Estes ambientes devem ser munidos de ferramentas que penalizem os utilizadores que apresentem comportamentos desviantes (egoísmo, falsificação de resultados, etc.). Normalmente, estes requisitos são endereçados por: $i)$ sistemas de incentivo, em que os dispositivos recebem uma retribuição pela execução de uma tarefa; ou $ii)$ sistemas de reputação que detetam dispositivos egoístas e os anunciam aos seus pares. Esta tese apresenta o sistema híbrido de reputação e incentivo (HRI), que combina as características de ambos os sistemas de incentivo e de reputação. O HRI aborda ainda o requisito da privacidade, uma vez que assume e encoraja os utilizadores a mudarem frequentemente o seu pseudónimo. O trabalho mostra que este sistema deteta várias formas de ataque, mesmo sem necessitar que os dispositivos estejam ligados à Internet, através da definição de diferentes modelos de ameaça onde mesmo com 90% de dispositivos egoístas na rede foi possível detetar 87% dos dispositivos egoístas.

Por não impor a conectividade a uma entidade confiável e centralizada que aplique de imediato as penalizações, o modelo poderia por em risco todo o ambiente, uma vez que encoraja os utilizadores bem comportados a assumirem também eles comportamentos desviantes ou a abandonar o sistema. Adicionalmente, o sistema é vulnerável a interpretações incorretas de comportamentos, que resultam quer das incertezas que caracterizam o meio de execução, quer de atuações maliciosas de alguns utilizadores. Para isso o modelo introduz uma entidade central confiável (ECC) que deve ser contactada esporadicamente por todos os dispositivos e que é responsável por disseminar a

informação de reputação dos dispositivos e por funcionar como fiel depositário do dinheiro virtual dos diversos dispositivos. Para tal, a ECC usa um algoritmo de predição de contactos que dissemina a informação de reputação dos dispositivos de forma enviesada, incluindo na lista enviada para cada dispositivo a lista dos dispositivos que são esperados ser encontrados num futuro próximo.

Uma das aproximações mais comuns para criar o algoritmo de predição é extrapolá-la a partir de experiências anteriores. Este trabalho investiga os padrões recorrentes de contactos observados entre grupos de dispositivos usando um conjunto de 9 anos de registos de acesso a uma rede sem fios, produzidos por 76479 dispositivos que acederam a um dos 239 pontos de acesso da rede eduroam existente no Instituto Politécnico de Lisboa (IPL). Este esforço permitiu modelar as probabilidades de ocorrência de um contacto, numa data predeterminada, entre grupos de dispositivos usando uma distribuição em curva de potência que varia de acordo com o tamanho da vizinhança e período de recorrência.

Em termos gerais, este modelo pode ser utilizado por aplicações que necessitam de disseminar grandes conjuntos de dados por grupos de dispositivos. A tese apresenta um algoritmo que fornece previsões de contactos diários, baseando-se no histórico de encontros entre pares e a sua duração, que foi posteriormente aplicado ao HRI. Este algoritmo conseguiu melhorar em quase 38% a capacidade do HRI detetar dispositivos egoístas. Adicionalmente este algoritmo pode ser utilizado para disseminar também informação sobre dispositivos não egoístas, mas com grande probabilidade de serem encontrados e assim reduzir o número de mensagens trocadas entre dispositivos para confirmação de reputação e resultados reduzindo assim a utilização de energia do ACMC. Esta variação, permitiu manter a capacidade de deteção de dispositivos egoístas nos valores anteriores e, permitir que quase 37% das trocas de tarefas entre dispositivos fossem otimizadas.

O mesmo conjunto de dados foi utilizado para identificar padrões de mobilidade humana. A compreensão destes é relevante para o desenvolvimento e avaliação de aplicações ubíquas. Têm vindo a ser desenvolvidos diversos modelos que permitem

contornar a escassez e as dificuldades existentes em capturar dados de mobilidade. A precisão na replicação da mobilidade humana observada por estes modelos varia. Maioritariamente, cada modelo concentra-se em replicar algumas das métricas que foram observadas, enquanto negligência outras. Infelizmente, todos os modelos tendem a descurar a diversidade, na função e objetivos dos utilizadores mas também nos dispositivos que são usados para aceder à rede sem fios. A tese introduz o MobIPLity, um gerador de cenários de mobilidade que extrai registos de movimento a partir dos registos de acesso presentes no conjunto de dados anteriormente referido. Os resultados do MoIPLity são tornados públicos na expectativa de que a sua escala permita suportar avaliações baseadas exclusivamente em dados de mobilidade reais, removendo assim a incerteza que surge da utilização de modelos de mobilidade sintéticos. Os registos mostram que existem diferenças entre tipos de dispositivos, com impacto em aspetos como a duração observada dos registos de mobilidade, velocidade, tempos de pausa e entre contactos e disponibilidade e que dificilmente é replicada por outros modelos de mobilidade sintéticos.

Os registos são comparados com modelos sintéticos utilizando um conjunto de métricas. A comparação levanta algumas questões relativamente às assunções feitas nos cenários típicos de avaliação com modelos sintéticos. Observou-se que o número crescente de *smartphones* resulta em mudanças significativas do padrão de utilização, com impacto na quantidade de tráfego e tempo de ligação dos utilizadores à rede. Esta analise permite identificar mudanças na forma de mobilidade dos dispositivos móveis ao longo dos últimos anos e fundamentar, assim, novas aplicações que aproveitam este extenso aglomerado de poder de computação e armazenamento recorrendo a formas de comunicação direta sem fios.

Abstract

Although the computational power of mobile devices has been increasing, it is still not enough for some classes of applications. In the present, these applications delegate the computing power burden on servers located on the Internet. This model assumes an always-on Internet connectivity and implies a non-negligible latency. Cloud computing is an innovative computing paradigm where the resources made available by a number of servers are transparently shared by its users. Cloud computing simplifies resource management, establishing the ground for the elastic computing concept, where each user can easily enlarge or reduce his amount of resources. The thesis studies the challenges and contributions posed to the application of the cloud computing concept to wireless networks. The goal is to define a reference architecture for high performance mobile application, the Collaborative Cloudless Computing (C3) together with a framework that addresses the challenges raised decomposing it on a series of components. The framework, named Mobile Collaborative Cloudless Computing Environment (MC3E) dismisses the connection to the Internet. In this scenario, computing power is obtained from neighbouring mobile devices, which coordinate to achieve a common goal: the execution of tasks requested by one or more participants. Expectations are that the use of the C3 concept contributes to: $i)$ reduce the response time, in comparison with delegations on commercial clouds; $ii)$ reduces user frustration when Internet connectivity is not available or its bandwidth is not sufficient; and $iii)$ alleviates the bandwidth consumed by these applications in the cellular infra-structure.

Allowing a mobile device to provide a service to the neighbouring peers carries non-negligible risks, of which confidentiality, privacy and selfishness are good exam-

ples. To discourage selfishness, two approaches are typically taken: *i)* in trade based systems, devices agree on a retribution for the execution of a task; *ii)* in trust based systems rogue devices are detected and advertised by their peers. This thesis describes and evaluates a hybrid system, combining trade and trust-based characteristics. We call this approach the Hybrid Trust and Trade system (HTnT). HTnT suits well privacy requirements as it assumes and encourages users to frequently change their pseudonyms. The work shows that the service detects several misbehaving approaches, even without requiring interacting devices to be connected to the Internet.

HTnT will use the capability to anticipate a contact with another device. This knowledge is useful for other applications that rely on some form of data harvesting or hoarding.

One of the most promising approaches for contact prediction is to extrapolate from past experiences. This work investigates the recurring contact patterns observed between groups of devices using an 9-year dataset of wireless access logs, produced by 76479 devices that connected to one of the 239 access points of the eduroam network at the Lisbon Polytechnic Institute (IPL). This effort permitted to model the probabilities of occurrence of a contact at a predefined date between groups of devices using a power law distribution that varies according to neighbourhood size and recurrence period.

In the general case, the model can be used by applications that need to disseminate large datasets by groups of devices. As an example, we present and evaluate an algorithm that provides daily contact predictions, based on the history of past pairwise contacts and their duration, that will be applied into HTnT.

Human mobility pattern analysis also used the same dataset. The understanding of human mobility patterns is key for the development and evaluation of ubiquitous applications. To circumvent the scarcity and difficulties in capturing mobility data, a number of models has been devised. The accuracy in replicating observed human mobility by these models varies. In general, each model concentrates n replicating some of the metrics that have been observed, while neglecting others. Unfortunately, all tend to

neglect diversity, in the roles and goals of the users but also in the devices that are used to access the wireless network. We present MobIPLity, a mobility scenario generator that extracts mobility traces from the access records of the IPL dataset. MobIPLity is made publicly available in the expectation that its large scale permits to support evaluations based exclusively on real mobility data, thus removing the uncertainty that emerges from the use of synthetic mobility models. Traces emphasise the differences that can be found between device types, with impact on aspects like the observed trace duration, speed, pause times, inter contact times and availability and which can hardly be replicated on synthetic mobility models.

The extracted mobility traces allowed for a comparison with other mobility models, where it was observable that the increasing number of smartphones resulted in significant changes to the utilization pattern, with impact on the amount of traffic and users connection time.

Palavras Chave

Redes móveis ad hoc,
Sistemas de reputação,
Modelos de mobilidade,
Predição de contactos

Keywords

Mobile ad hoc networks,
Reputation systems,
Mobility models,
Contact prediction

Agradecimentos

Ao Prof. Hugo Miranda que, além de todo o empenho na orientação, foi um modelo ímpar na sua capacidade de trabalho, empenho e motivação. Foi também um modelo para a minha atividade profissional: espero saber ter, com os meus alunos, a mesma capacidade de orientação e liderança com que tive a sorte de lidar diariamente nos últimos anos.

Ao Prof. Pedro Veiga quero agradecer toda a disponibilidade na fase inicial do meu projeto de doutoramento. Foi um pilar na minha formação.

Ao IPL, aos meus colegas da IPLNet e em particular ao Eng. Pedro Ribeiro quero agradecer todo o apoio prestado, nomeadamente no acesso ao conjunto de dados utilizado nesta tese sem o qual dificilmente conseguiria concluir este trabalho.

Ao Gloss/LaSIGE quero agradecer o apoio financeiro que me permitiu apresentar a maioria dos trabalhos em conferências.

À Susana e à Clara pela confiança e alento contínuas, que em alturas de maior desgaste me permitiram levar este trabalho até ao fim.

Aos meus pais, Maria e Manuel Cruz, um especial obrigado por todo o apoio, paciência, sacrifício e por me apoiarem desde sempre, num esforço contínuo.

A todos, sem vocês nada disto seria possível.

Lisboa, Maio de 2015

Nuno Miguel Machado Cruz

À Susana

À Clara

Contents

List of Figures

List of Tables

Acronyms

ACL Access Control List

AP Access Point

ART Android RunTime

C3 Collaborative Cloudless Computing

CCDF Complementary Cumulative Distribution Function

CD Consecutive Day

CMD Consecutive Month Day

CTE Central Trusted Entity

CTE Central Trusted Entity

CTE Central Trusted Entity

CW Consecutive Week

CWD Consecutive Week Day

DCT Discrete Cosine Transform

DHCP Dynamic Host Configuration Protocol

GCM Google Cloud Messaging

GPS Global Positioning System

HDFS Hadoop Distributed File System

HTnT Hybrid Trust and Trade

ICT Inter-Contact Time

IPL Lisbon Polytechnic Institute

KDE Kernel Density Estimations

MAC Medium Access Control

MAI Application Interface Module

MANET Mobile Ad Hoc Network

MC3E Mobile Collaborative Cloudless Computing Environment

MSAS Mobile Service and Application Store

NBS Nash Barganing Solution

NM Node Manager

OUI Organizationally Unique Identifier

PDA Personal Digital Assistant

PDF Probability Distribution Function

PKI Public Key Infrastructure

RFM Rank of the First Miss

RPC Remote Procedure Call

RWP Random WayPoint

SC Service Container

SFM Score of the First Miss

SMD Small Mobile Devices

SMI Structure of Managed Information

TC Temporal Communities

TMS Trust Managment Server

TPM Trusted Platform Module

TSC Temporal Sub-Community

UDP User Datagram Protocol

WNA Weighted Network Analysis

1

Introduction

The increasing number of mobile devices[1] (smartphones, tablets, laptops, etc.) combined with informal observations of their usage pattern suggest that, in locations where a significant concentration of individuals exists (shopping malls, cultural or sport events, public transportation, vehicles), it is possible to find unused resources (such as CPU cycles) on neighbouring devices, possibly carried by other users. Available resources are directly tied to the ever increasing capabilities of the mobile devices. In spite of this aggregated amount of resources, applications frequently resort to the Cloud, using an Internet connection, in order to expand their computational power. This approach facilitates the development of resource hungry applications, such as voice recognition (used on personal assistants such as Siri, Cortana or Google Now) or route calculation (as in Apple Maps, Google Maps or Bing Maps).

This thesis proposes the Collaborative Cloudless Computing (C3) concept. In C3, computing power is obtained from a federation of mobile devices in proximity. Members coordinate to achieve a common goal: the execution of tasks requested by one or more devices using exclusively the resources made available by the federation. C3 is not expected to execute all tasks. Good examples are image or audio processing, or any other task that uses data already present on the devices or acquired by the users. Expectations are that the use of the C3 concept contributes to: i) reduce the application response time, in comparison with delegations on commercial clouds; ii) reduce user frustration when Internet connectivity is not available or its available bandwidth

[1]See "Smartphone Market Share" Avaliable at: http://www.idc.com/prodserv/smartphone-os-market-share.jsp (Last access on: 20 April 2015)

is not sufficient; $iii)$ improve energy efficiency by reducing energy consumption of the cellular network interface by in turn use the less power hungry short range wireless interfaces; and $iv)$ reduce bandwidth impact in the cellular network infrastructure (e.g. 3G and 4G), already overloaded in places where a large concentration of mobile device exist (Tan et al., 2008; Wortham, 2009). Bandwidth gains can be achieved by executing the task locally, or by sharing task results. Result sharing, is something to consider since it has been shown that the likelihood of finding common tasks among neighbouring devices is high (Kangasharju et al., 2010).

The contribution of this work is justified by the following case studies:

Alice is on vacations in Turkey, and is currently on an excursion organized by a national travel agency, visiting the ruins of the ancient city of Ephesus. At the entrance an audio-guide was provided. Unfortunately, Turkish was the only available option. Since Alice isn't a fluent Turkish speaker, she will try to use the audio translation application from her smartphone. However, her Internet connection has a very low quality, the latency and available bandwidth aren't enough to provide a real-time translation. Since there are more users in the same excursion that are interested in the same translation, the mobile devices organize between themselves to obtain the computational power needed for real-time translation and all the users can listen the audio-guide notes in their native language.

In another museum, Alice finds that some work of art has a description in Turkish. She uses the camera from her device to take a photo of the description to be translated to her native language. Unable to locally obtain the computational power needed for the translation from her smartphone, and given the absence of Internet connectivity in the region, the application uses C3 to obtain the computational powered needed. The translation result is itself stored in C3, becoming available to other users.

Alice and her son Tommy are at the amusement park when all of a sudden, Tommy disappears. After calling for help, Alice uses the emergency search app on her cell phone to spread a picture of Tommy that she had taken earlier. An alarm with the picture is gossiped in the background by the smartphones at the park, who start to look for Tommy in photographs shot by the devices recently. However, because face recognition is CPU intensive, smartphones extract

faces from the photographs and delegate the facial recognition process to other smartphones in the neighbourhood with spare CPU time, effectively sharing the load and speeding up the process. In addition, some of the smartphones alert their users, thus creating a surveillance net that increases its diameter every 5min. The picture is equally delivered to autonomous robots in the park who use real time face recognition software to try to find Tommy. Chloe, an autonomous robot who has received the picture, recognises Tommy and notifies Alice through the emergency app, telling her where to pick up Tommy.

1.1 Problem Statement and Objectives

The introduction of the C3 model raises some new issues that cannot be found when the traditional cloud provider model is used. On the C3 environment applications use an untrusted source for computation, which is affected by the inherent rational behaviour of mobile device owners. Rational behaviour can assume two perspectives: $i)$ participants may require some sort of incentive in order to use the platform and share their own mobile device resources; $ii)$ participants may use the platform to exploit others' devices without sharing their own resources, resulting in a longer lifetime of the own mobile device battery, at expenses of the remaining.

To encourage the adoption of the C3, an incentive has to be provided to keep devices active and trust has to be established between peers with an acceptable behaviour. However, the creation of incentive and trust on the C3 must have a reduced impact on the system as it must consider the limited resources of the devices. Therefore, research of reputation systems, digital coin and dissemination algorithms must be revisited to consider resource and connectivity limitations, such as reduced local storage, energy or Internet connectivity. The dissemination of trust information will, in our envisioned scenario, play a fundamental role and can benefit from a good estimate of contact prediction between devices, to alleviate the device storage. An ideal implementation of C3 would be able to predict contacts between devices and create a trust and incentive ring between them, to allow the sharing of resources to occur

when needed. Incentive can be materialized like a trade market, where resources are exchanged between devices ensuring fairness.

This thesis investigates some of the problems posed to the successful deployment of a system inspired by the C3 model. In particular, it aims to design and implement a system that enables trust, by detecting misbehaviour, and incentive, rewarding resource sharing, in order to enable the adoption of the C3. This system will keep the resource usage on each device low by using strategies based on contact prediction. The contact prediction algorithm is created by observing contacts between devices on a large dataset of Wireless LAN (frequently referred as WiFi) access records.

1.2 Contributions

The main contributions of this thesis are:

- The definition of an architecture for Collaborative Cloud Computing (C3), where devices harvest resources from nearby neighbours and make available resources for demanding applications to the users regardless of Internet connectivity.

- A system that supports incentive and trust metrics between mobile devices. Incentive is responsible for keeping users motivated for sharing their local resources. Trust is established between devices that behave correctly, while identifying possible misbehaving actions from others by creating and sharing a set of reputation information from/to all devices.

- An algorithm for prediction of contacts between peers, able to improve the dissemination of reputation information by anticipating devices that will be in range in a nearby future.

1.3 Results

The thesis presents the following results:

- A mobility scenario generator that creates mobility traces using real world data. This scenario generator improves the evaluation process quality by using actual data instead of the output of a statistical model.
- An extensive analysis of mobility data from the access records of 76479 mobile devices that accessed at least one of the 239 access points of the eduroam WiFi network of the Polytechnic Institute of Lisbon between 2005 and 2013.
- An extensive analysis on the recurrence of contacts between temporal communities of peers on a large dataset of mobility data together with a statistical modelling of the characteristics observed.
- A hybrid trust and incentive reputation system for a C3 environment that uses a contact prediction algorithm to disseminate information about misbehaving devices to peers in a nearby future.

1.4 Outline of the Thesis

The thesis is structured as follows. Chapter 2 motivates the problem, presenting the system model and discussing the multiple challenges raised by the C3 concept.

Chapter 3 presents the related work, which reflects the multiple lines of research involved on the C3 concept. Each line of research is supported by a discussion of some of the most relevant works in the area.

Figure 1.1 presents the outline of the following chapters of this thesis. The system model and C3 architecture are detailed in Chap. 4. The chapter raises a discussion on the challenges and possible solutions that can be pursued. For each challenge/solution a set of components is proposed in order to define the C3 framework.

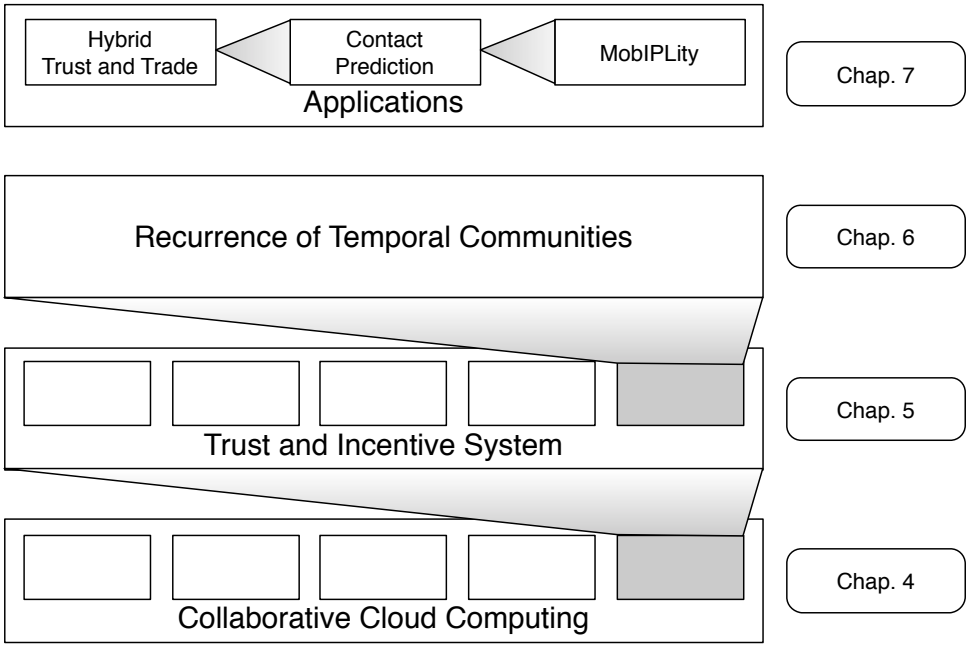


Figure 1.1: Thesis outline

The component responsible for the trust and incentive, in particular a digital cash protocol, are presented on Chapter 5.

The trust and incentive system uses the knowledge obtained from the analysis of a large dataset of Wi-Fi access records to create an algorithm for predicting the recurrence of temporal communities between mobile devices, which is the focus of Chap. 6.

In Chap. 7, the Hybrid Trust and Trade system implementation is detailed, which address the trust and incentive requirement and uses the knowledge previously obtained to make predictions of contacts between devices. The evaluation of such algorithm uses mobility traces that were generated by MobIPLity, our mobility scenario generator, and other mobility traces that are publicly available.

Chapter 8 summarizes this document with most significant conclusions and directions for future lines of work.

Related Publications

Preliminary versions of portions of this dissertation have been presented in the following publications:

Papers at international conferences or journals (all the following are available on ACM and/or IEEE digital libraries):

- Cruz, Nuno, and Miranda, Hugo. 2015. Recurring contact opportunities within groups of devices. In *Proceedings of the 12th International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services*, MOBIQUITOUS '15, ICST, Brussels, Belgium, Belgium. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), Coimbra, PT. 2015. To appear.

  In this paper we present an algorithm for prediction of contacts between devices using the knowledge of contacts observed in the eduroam WiFi network access logs from the Polytechnic Institute of Lisbon from 2005 to 2013. We applied this data a dataset of taxi movements in Rome to evaluate the prediction capabilities across multiple environments.

- Cruz, Nuno, and Miranda, Hugo. 2014. MobIPLity: A trace-based mobility scenario generator for mobile applications. In *Proceedings of the 11th International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services*, MOBIQUITOUS '14, pages 228-237, ICST, Brussels, Belgium, Belgium. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), London, UK. 2014.

  This paper presents the mobility scenario generator that uses the eduroam dataset as input. Additionally, a web interface is presented on this publication, allowing anyone to create mobility scenarios based on our dataset. An extended version of this work is expected to appear in:

- Cruz, Nuno, and Miranda, Hugo. 2015. MobIPLity: A trace-based mobility scenario generator for mobile applications. *EAI Endorsed Transactions on Ubiquitous Environments*. 2015. To appear.

- Cruz, Nuno, Miranda, Hugo, and Ribeiro, Pedro. 2014. The Evolution of User Mobility on the Eduroam Network. In: *2014 IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*, pages 249–253, Budapest, Hungary. 2014.

  The eduroam dataset is extensively studied in this paper, presenting a statistical analysis and popular metrics on the features present on the dataset. The work is mainly focused on the year of 2012.

- Cruz, Nuno. (2014). Mobile collaborative cloudless computing. In *2014 IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*, pages 184–186, Budapest, Hungary. 2014.

  The C3 concept was presented on this work. This work was presented at the PhD Forum of PERCOM 2014.

- Busnel, Yann, Cruz, Nuno, Gillet, Denis, Holzer, Adrian and Miranda, Hugo. (2013). Reinventing mobile community computing and communication. In *2013 12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, pages 1450–1457, Melbourne, Australia. 2013.

  This vision paper presents a new perspective on an application for mobile devices that is supported by the C3.

- Cruz, Nuno, and Miranda, Hugo. (2013). A Hybrid Trust and Trade Service for Mobile Collaborative Computing. In *2013 Seventh International Conference on Next Generation Mobile Apps, Services and Technologies (NGMAST)*, pages 1–6, Prague, Czech Republic. 2013.

  The trust and incentive system for the C3 is presented in detailed in this work. Evaluation was done using an implementation of the complete system in the OMNet++ Simulator using multiple threat models.

National (Portuguese) conferences (available on conference website):

- Cruz, Nuno, Miranda, Hugo, and Ribeiro, Pedro. (2014). O impacto dos smartphones nos modelos de mobilidade tradicionais. In *Atas do 6o Simpósio de Informática (INForum 2014)*, pages 195–210, Porto, Portugal. 2014.

  This work presents the mobility scenario generator, focused on the comparison with current mobility models using popular metris. The paper exploits the weaknesses in current mobility models not considering observed differences between smartphones and other mobile devices.

- Cruz, Nuno, Miranda, Hugo, and Ribeiro, Pedro. (2013). A mobilidade dos utilizadores da eduroam ao longo dos anos. In *Atas do 5o Simpósio de Informática (INForum 2013)*, pages 189–200, Universidade de Évora, Portugal. 2013.

  The mobility of the users on the eduroam WiFi access records of 2013 is presented on this work.

- Cruz, Nuno, and Miranda, Hugo. (2012). Avaliação de um sistema de reputação e incentivo. In *Atas do 4o Simpósio de Informática (INForum 2012)*, pages 144–155, Faculdade de Ciências e Tecnologia da Universidade Nova de Lisboa, Portugal. 2012.

  The trust and incentive system was first detailed in this publication that presented results based on an evaluation that used an implementation of the algorithm in Java.

- Cruz, Nuno, and Miranda, Hugo. (2011). Arquitectura para uma Computação em Nuvem Colaborativa entre Dispositivos Móveis. In *Atas do 3o Simpósio de Informática (INForum 2011)*, pages 450–455, Coimbra, Portugal. 2011.

  This short vision paper presented our first vision of the C3, raising the problems of such an architecture.

2

Motivation

Several popular applications for mobile devices require an Internet connection to provide their services. This requirement is supported by the need to access Cloud resources, normally computing power and storage. This limitation could be circumvented if the same resources were provided by a framework that explores the Collaborative Cloudless Computing (C3) concept. In addition, this approach would leverage a better user experience and, simultaneously reduce the impact on battery usage and improve broadband cellular bandwidth consumption.These devices are personal to the user and carry sensitive information that cannot be disclosed to other participants. In addition, participants may not share a common goal. Although nodes are expected to frequently enter or leave this network, it is expected that at any instant the nodes in range can provide the resources required for a service that is currently delegated to a cloud infrastructure by an application.

This system model rests on the assumption of the availability of a non-negligible mobile device "density", supported by: $i)$ the frequent reports on the fast expansion of the smartphones market[1]; $ii)$ by noticing that in some locations (e.g. trains, sport events at large stadiums), mobile devices are expected to remain in proximity for reasonable amounts of time (Cruz et al., 2014); and $iii)$ by observing that in most of the time, mobile devices remain idle (Karlson et al., 2009).

[1]See "Smartphone Market Share" Avaliable at: http://www.idc.com/prodserv/smartphone-os-market-share.jsp (Last access on: 20 April 2015)

2.1 Challenges

The proposed system model raises a number of challenges on the implementations of C3s. The remainder of this section lists a few of them and addresses, in a critical perspective, some existing strategies for their resolution, that can be found in the literature.

2.1.1 Privacy

Privacy and integrity of user data are fundamental for the acceptance of C3 by users. For those providing resources, C3 must prevent the leaking of personal data. The concept of personal information should be interpreted in a wide sense of the word. That is, beyond classic examples of personal data contained in the device, it is also important to conceal information that would allow the association between user and device, for example, by observing the existence of repetitive patterns of user presence in a particular location. To the resource consumer, C3 must ensure privacy of data delegated to other devices for processing.

2.1.2 Malicious Behaviour

Malicious behaviour can be expressed in a variety of ways, for example by never making any resources available to the community by being selfish or by returning bogus, effortless results to others requests. One node's selfish behaviour can threaten the effective deployment of C3s, as it can motivate others to reply with a similar behaviour. The C3 must be responsible for detecting and punishing malicious behaviour, by refusing to accept tasks from rogue devices. However, C3s must have a memory that allows good behaviour to be compensated in the future.

2.1.3 Cost of Task Division

The delegation of tasks on other devices consumes computing power, storage, bandwidth and battery. To be useful, C3s must attenuate these costs, rewarding with a lower latency than the one available while using the original cloud provider. One approach is to eliminate the need to transfer code between devices, limiting the collaborative computing to generic computational blocks integrated in the platform itself. Examples are complex arithmetic functions or voice recognition. The challenge is to determine a suitable complexity of computing blocks. Simple computing blocks are easier to deploy, allowing more fairness, lower battery consumption during computation and tolerating user interruption without a significant impact on execution. Complex computing blocks are more useful to applications and require a smaller number of tasks to achieve a result. Network transitivity has more impact on complex computing blocks, considering that on some occasions the task requester becomes unreachable, implying that the results of the task executions are lost. Meanwhile, in a hybrid environment, the computing blocks concept could also be expanded to nearby cloud nodes.

2.1.4 Computing Block Location

The distribution of the computational effort requires knowledge, by the device that performs the request, of what computing blocks are available in the neighbourhood. Computing blocks are a pre-defined component that specifies task or set of tasks that are offered by a peer device. The advertisement of available computing blocks must consider the shared physical resources, such as local storage, battery and CPU shares of the devices and must be economical, avoiding waste of battery and bandwidth.

2.1.5 Energy Consumption

All tasks, and the C3 itself, imply a certain amount of consumed energy. On mobile devices the available energy is restricted to the available battery capacity. As such,

there's the need to determine if the delegation of a task to the C3 has lower energy footprint than the one obtained while using the traditional cloud services. Similar research already exists on determining when to offload to the cloud, however further research is needed to determine the applicability of these strategies on the C3.

3

Related Work

This study of the related work is organised as follows. Section 3.1 identifies the most relevant components of the Collaborative Cloudless Computing (C3) model and the equivalent counterparts found in research. In Sec. 3.2 the related work for the incentive and trust system is discussed. In the following sections (Sections 3.3 and 3.4) the related work for the components needed to achieve a dissemination algorithm that predicts contacts between devices is presented.

3.1 Collaborative Cloudless Computing

Cloud computing is the paradigm of sharing resources between companies that outsource their IT infra-structure to a third party in order to reduce operational costs. The sharing of resources brings costs benefits due to savings obtained by removing the need of a locally managed infra-structure, allows elasticity for organizations by enabling resources to be allocate by demand, and applications to use a number of resources that exceeds the ones available locally. We consider this the traditional cloud as opposition to newer forms of cloud computing, as will be later depicted.

The traditional cloud model can be categorized into three classes, distinguished by the layer at which resources are shared: Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS) and Software-as-a-Service (SaaS) (also known as Application-as-a-Service) (Mell and Grance, 2009; Armbrust et al., 2010). IaaS delivers computer infrastructure to its users, typically abstracted above a layer of virtualization technology.

PaaS supplies a computing platform and a solution stack as a service; this uses an already existing cloud infrastructure, and allows customers to build their applications directly on top of the platform. In SaaS complete applications are provided. A typical example is a CRM (Customer Relationship Management) application or an office suite, like the one provided by Google Docs.

Alternatively, clouds can also be classified by the way users access them. Public clouds are available to anyone. In other words, anyone can become a customer of a public cloud provider. Private clouds are internal to an enterprise. Community clouds are clouds that are shared between companies, typically to reduce operational costs.

3.1.1 Mobile Clouds

Mobile cloud is a term that depicts the use of the traditional cloud, by mobile applications in an effort to obtain additional computational power, storage space, or other available resource. These applications can be split into different categories according to the requested resources. Applications that require a higher computational power typically execute a task on a cloud provider using data that was acquired by the device or provided by the user. Examples of these applications are personal assistants, such as Siri (Aron, 2011), that send the user's voice to the cloud in order to provide a virtual personal assistant service, or Google Voice (Schalkwyk et al., 2010) that allows search by voice. Similarly, Google Goggles[1] allows a user to search the web or translate text using an image taken by the mobile device camera. The second class of mobile applications that use the cloud are applications that resort to the cloud to access a large dataset of data. Examples are mapping applications such as Google Maps,[2] Apple Maps[3] or Microsoft Maps.[4]

[1]See "Goggles App" https://play.google.com/store/apps/details?id=com.google.android.apps.unveil (Last access on: 20 April 2015)

[2]See "Google Maps App" at: https://play.google.com/store/apps/details?id=com.google.android.apps.maps (Last access on: 20 April 2015)

[3]See "Apple Maps App" at: https://www.apple.com/ios/maps (Last access on: 20 April 2015)

[4]See "Microsoft Maps App" at: http://apps.microsoft.com/windows/en-us/app/maps/97a2179c-38be-45a3-933e-0d2dbf14a142 (Last access on: 20 April 2015)

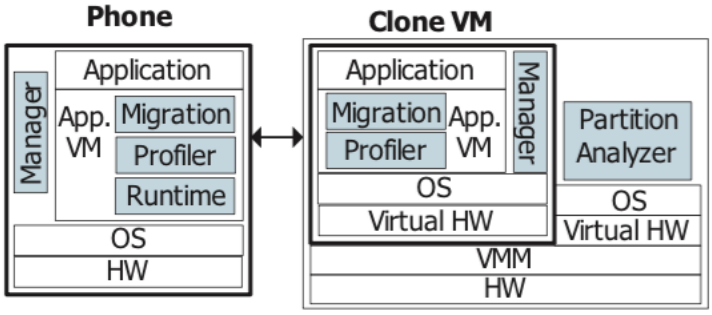


Figure 3.1: Clone Cloud Architecture (Chun et al., 2011)

The interested reader can read an exhaustive survey of mobile cloud computing architectures in (Fernando et al., 2013).

Theoretically, any mobile device, or group of mobile devices, with networking capabilities can perform the role of a cloud node and there are multiple lines of research investigating this approach. One distinctive element is the geographical distance from the application to the cloud infra-structure, that can range from the traditional cloud, far from the application, to a cloud where mobile devices become the cloud itself, thus much closer to the application.

Cloudlets (Satyanarayanan et al., 2009), fog computing (Bonomi et al., 2012), pervasive computing (Satyanarayanan, 2001), opportunistic computing (Conti et al., 2010), crowd computing (Murray et al., 2010), and cooperative computing (Borcea et al., 2002), are all architectures where the cloud or computing resources are pushed closer to the user. Resources can be provided by a specialized device, an access point with more resources, or another device. One of the motivations behind these architectures is the attenuation of the negative impact posed by the latency and network connectivity to commercial clouds. All, address latency, geographical location of the data and privacy concerns, by providing storage, computing or more specific application services closer to the user. Both fog computing and the cloudlets concepts anticipate a market of resources, whose openness needs to be assured, in order to prevent proprietary software ecosystems, as discussed recently by the author of cloudlets in (Satyanarayanan et al., 2015).

CloneCloud (Chun et al., 2011) is an example of a mobile cloud platform that implements the cloudlets and fog computing model. CloneCloud leverages deployment of applications to the cloud by delegating intensive computing operations to clones of mobile devices. These clones run the Dalvik VM (that was recently superseded by Android ART (Google, 2015)) on hardware deployed in the wired infrastructure. These clones are full images of the mobile device that run augmented applications. Figure 3.1 depicts CloneCloud architecture. CloneCloud exploits the existing hardware abstraction of the Android opensource operating system to expand application support to cloud node functioning on other hardware platforms.

CloneCloud authors envisioned many types of application augmentation (running virtualized copies on the cloud) (Chun and Maniatis, 2009). Primary augmentation is aimed to computation hungry applications, such as speech and video processing, leaving the interface to the mobile device, and the processing itself to the cloned device. Background augmentation copes with processes without a user interface, such as a file scan. In hardware augmentation, the clone device allows applications to use locally available resources, instead of mimicking the hardware limitations of mobile devices (e.g. memory), thus increasing application performance. Multiplicity augmentation uses multiple copies of the system image to increase the performance of parallel applications.

Changes on devices must be synchronized among clones trough a manager process (depicted in Fig. 3.1). When compared with the C3 model, CloneCloud assumes that the cloud nodes are trusted. The authors use a cost model to determine if the application should be run on the device or have its execution sent to a nearby clone. The cost metrics considered the execution time and energy consumed at the mobile device, that if above a certain threshold triggers the process of outsourcing the application to a clone.

MAUI (Cuervo et al., 2010) takes a similar approach to CloneCloud, using nearby hardware as a cloud node. However, MAUI cost model further involves application developers by soliciting method annotations to hint MAUI when to offload compu-

tation to a nearby cloud node. MAUI authors claim energy savings of up to 90% for using intensive face recognition applications. MAUI, as CloneCloud, also includes a cost-benefit analysis that profiles each applications' methods through serialization, combining it with run-time execution conditions like the available bandwidth and latency.

3.1.2 Mobile Collaborative Cloudless Computing

Mobile Clouds, can also use neighbouring mobile devices as cloud nodes, by harvesting the unused resources. In the literature, this specific case of mobile cloud computing assumes names as crowd computing and cooperative computing. It depicts an architecture where the mobile devices are used in an ad hoc topology for cloud computing. However, none of them manages to include the Cloud and Collaborative nature of the architecture as proposed by Mobile Collaborative Cloudless Computing.

Authors of (Huerta-Canepa and Lee, 2010) identified the following features to be expected on Mobile Collaborative Cloud Computing:

- Resource monitoring, to determine if a task could be executed locally.
- Integration with existing Cloud APIs.
- A partition and offloading scheme of tasks suited to the capabilities of mobile devices.
- Activity detection, to find users with the same goals.
- Discovery and selection of mobile devices by the underling network architecture.
- Memory cache scheme.
- Lightweight and resource friendly architecture.

Figure 3.2 presents the stack used to create the mobile cloud computing infrastructure proposed by the authors to address these requirements. The Application

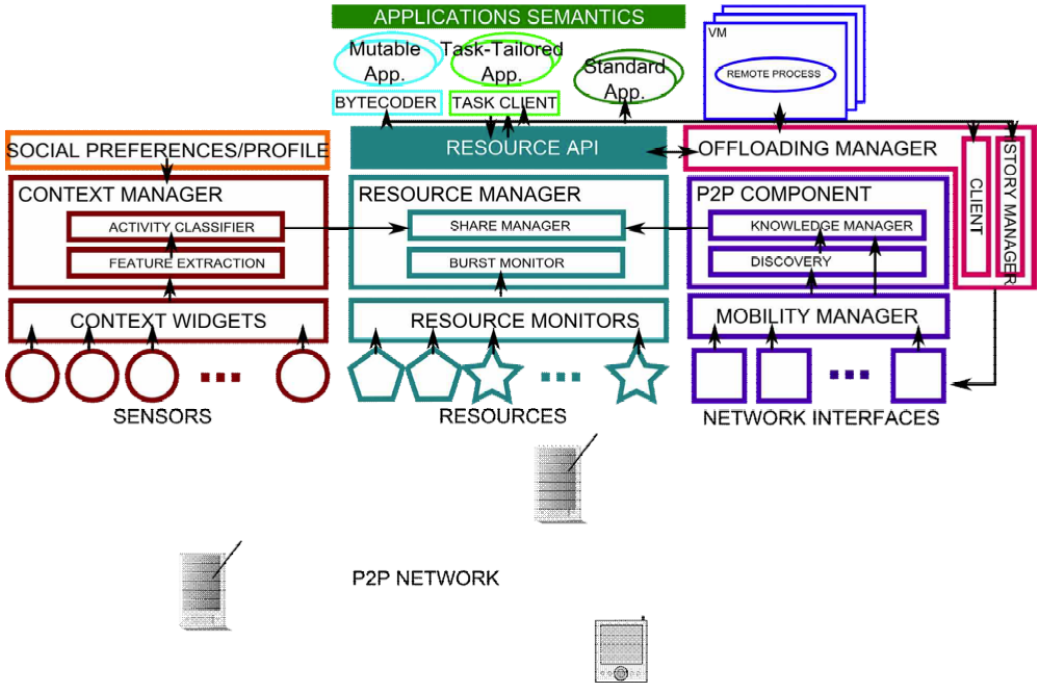


Figure 3.2: A Virtual Cloud Computing Provider for Mobile Devices architecture (Huerta-Canepa and Lee, 2010)

Manager, integrated in the Applications Semantics block, intercepts the application, modifying it to add the features required for offloading to a local cloud instead of a provider/infrastructure-based cloud. Resource Manager is responsible for profiling the application and monitoring the resources of the local device. Context Manager contains at least two basic contexts: location context (for mobility traces) and nearby devices context (for neighbourhood discovery). Nearby devices context is used by the P2P component. This component uses an ad hoc discovery mechanism and notifies context manager if new devices enter the surrounding area. This information is also used to detect if users are stable on a position. The offloading Manager component is responsible for task distribution and reception from nearby devices.

The previous work is also based on the Android virtualization capacity, in line with CloneCloud and MAUI, introducing mobile devices as cloud nodes, distributing computation and data among them.

A platform that also explores devices as cloud nodes, is Hyrax (Marinelli, 2009), which runs Hadoop (White, 2009) as a distributed processing framework to distribute

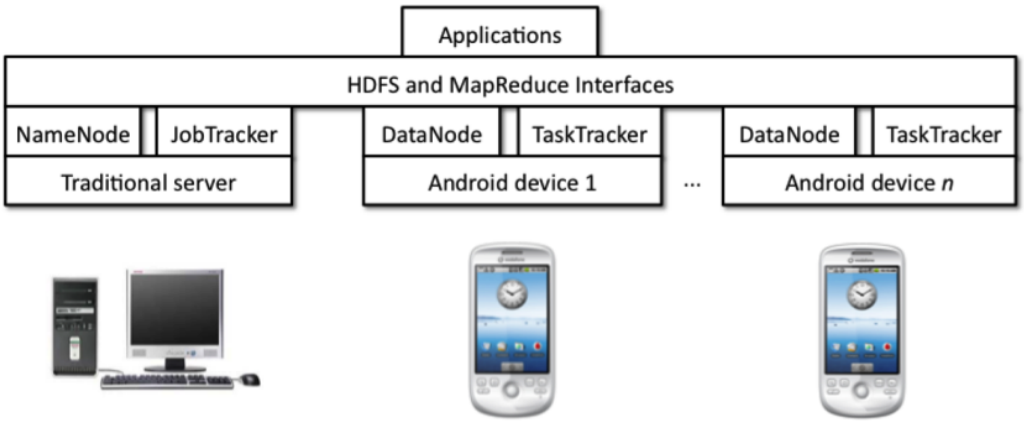


Figure 3.3: Hyrax Architecture (Marinelli, 2009)

tasks among other devices. Hadoop was also the selected platform by the authors of the previous work (Huerta-Canepa and Lee, 2010).

Hadoop is an Apache Foundation open source implementation of MapReduce (Dean and Ghemawat, 2008). MapReduce is a programming model and an associated implementation for processing and generating large datasets, developed by Google. MapReduce is highly scalable and fault tolerant. To use MapReduce, users specify a ”map” function that has a key/pair input and outputs intermediate key/value pairs. The user also specifies a ”reduce” function that processes each intermediate key/value pair and generates an output. MapReduce uses a distributed file system (the Google File System (Ghemawat et al., 2003)) to read inputs and write outputs. In contraste with MapReduce, Hadoop uses the Hadoop Distributed File System (HDFS) for this task.

Figure 3.3 presents the multiple components of Hyrax, shared with Hadoop. NameNode process maintains a directory of data blocks that make up the files in HDFS. TaskTracker executes tasks and JobTracker coordinates tasks among TaskTrackers. DataNode stores and provides access to HDFS data blocks.

Authors have identified that Hadoop would require considerable changes to be useful in mobile environments. In particular, it was observed that Hadoop uses several technologies that are CPU intensive such as XML parsing and servlets, requires

considerable memory and an performs an excessive number of I/O operations. The same conclusion was supported by the previous work (Huerta-Canepa and Lee, 2010). During its evaluation, it was determined that executing a task locally was more efficient than deploying it to the collaborative cloud. Authors of both works determined that Hadoop lacked the efficiency needed to support this environment.

Although the two previous architectures share the global principles defined for C3, both disregard selfishness and privacy issues, neither consider support to reduce latency with code transfer. This limits their application to scenarios where the users share a common objective and has a negative impact in the performance and battery lifetime.

3.1.3 Hybrid Platforms

The literature has also investigated hybrid architectures, where resources can be obtained either by the traditional cloud, or by neighbouring mobile devices. MobiCloud (Huang et al., 2010; Xing et al., 2013) aims to build a secure cloud framework for mobile computing. MobiCloud allows for cloning a complete device to the cloud (thus adopting an approach similar to MAUI or CloneCloud), but also considers a partial cloning to a nearby mobile device. MobiCloud addresses the security and privacy of the cloud nodes by integrating cryptography based solution that provides content privacy, and an authentication/authorization/accounting scheme. Data access control isolation in the multi-tenant scenario (a tenant in this context is an application running on a foreign mobile device) is identified as a privacy concerns in MobiCloud, and two control mechanisms are defined:

- Implicit Filter Based Access Control Isolation: When a shared resource is requested by a tenant application, this is done using a group key management based solution, where multiple virtual applications using the same physical device, share the same key.

- Explicit Permission Based Access Control Isolation: In this case a typical Access Control List (ACL) is used to explicitly specify the application access to shared resources.

These controls are visible on the architecture of MobiCloud, depicted on Fig. 3.4. The "virtual trusted and provisioning domain (VTaPD)" isolates security domains using network isolation between the infra-structure cloud nodes and mobile devices. These are controlled by the VTaPD Manager and Trust Managment Server (TMS). MobiCloud defines this new class of cloud service as Security-as-a-Service (SeaaS). Security in MobiCloud is based on Attribute Based Encryption (Bethencourt et al., 2007) for data encryption and decryption, where each attribute has multiple secret components for different users. Users can share an attribute, but the private keys for each attribute are different. Private keys, as opposing to a Public Key Infrastructure, are generated for each user according to his public attributes.

Software agents (SA) that provide services/applications to the mobile devices are contained in VTaPDs. Multiple SAs exist inside the same Service Container (SC) and are controlled by a Node Manager (NM) process. SCs are available on the traditional cloud and on mobile devices. Services/Applications are exported through the Application Interface module (MAI) to the mobile devices. Software agents are distributed through the Service and Application Store (MSAS).

MobiCloud assume that devices participate in a delay tolerant Mobile Ad Hoc Network (MANET) where devices can be used as routing hops in order to extend the cloud infra-structure, and where device parameters such as battery level or CPU power and networking parameters like bandwidth or delay are considered for routing decisions. Unfortunately, and despite considering the MobiCloud features addressing security and privacy of user data, MobiCloud does not address the trust and incentive problems.

SCAMPI (Pitkänen et al., 2012) is an EU project aiming to develop a platform for opportunistic computing where users can share resources. In SCAMPI, the human

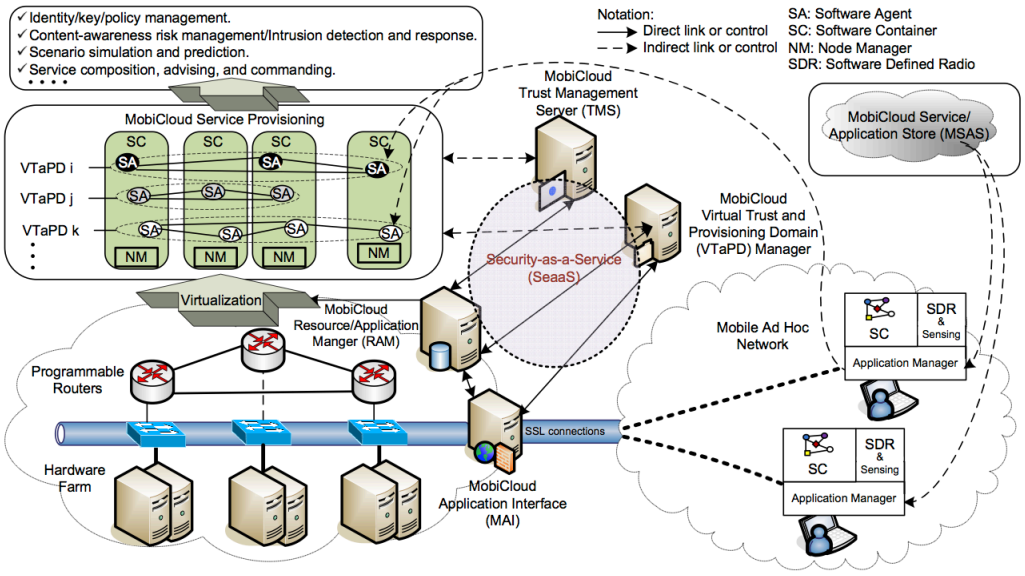


Figure 3.4: Mobicloud Architecture (Huang et al., 2010)

social behaviour forms a layer of research tightly connected with available resources (on Fig. 3.5). These can be obtained from peer devices and fixed installations, such as an access point. SCAMPI authors consider that users are also in charge of providing the content, than can be a publication. For this, they established a recommendation system, where peers evaluate the relevance of the content.

SCAMPI authors observed and characterized human mobility in order to evaluate resource stability, determining the efficiency of deploying tasks to peer devices. To improve the mobility prediction, in SCAMPI, authors studied the relation between online social networks and physical contacts. Due to the lack of datasets that contained both types of data, authors used current social networks, such as Facebook, to obtain social ties between users. However, the experimental study using a Facebook application only lasted three weeks and only included 22 individuals that were active during the study. Despite SCAMPI having as an objective an architecture sharing the goals of C3, the project produced more results on the human mobility component, a topic to be addressed later in this chapter.

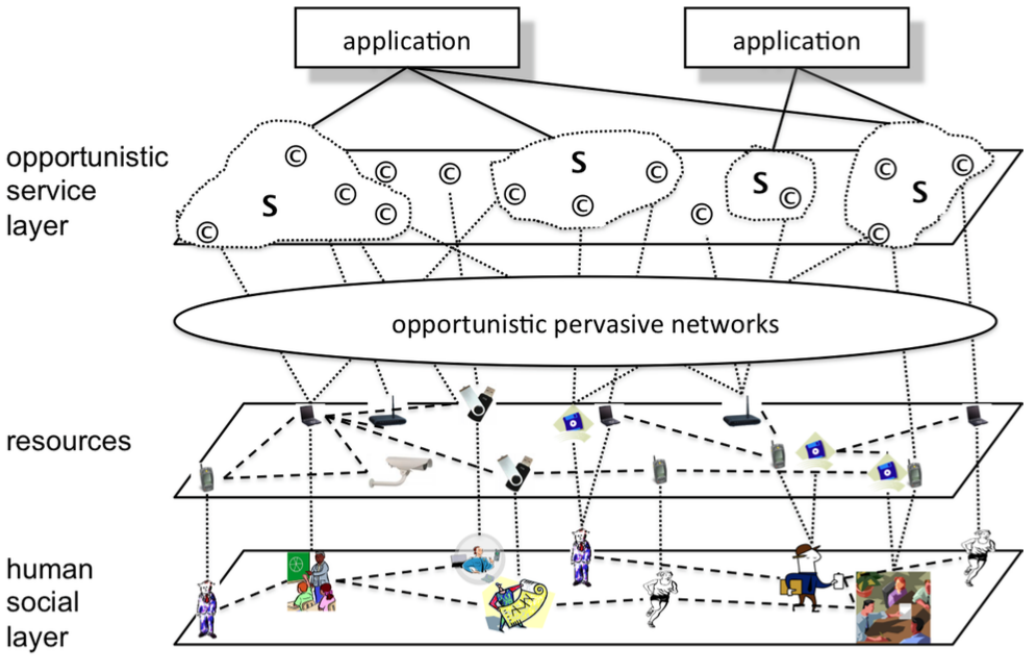


Figure 3.5: SCAMPI Architecture (Pitkänen et al., 2012)

3.2 Trust and Incentive

The adoption of the Collaborative Cloudless Computing (C3) requires that users share their resources in order to use others' resources for their own tasks. Game theory suggests that players of a given system behave rationally. From all possible actions, the players will choose the one that brings the highest cost-benefit ratio (Neumann and Morgenstern, 1944). In the C3 environment this is portrayed in the devices' decision of sharing their resources. If not otherwise encouraged, users' will only share their device resources if some benefit is attained.

Measuring the degree of participation of the peers in collaborative environments has usually been implemented with either trust or incentive/trade based approaches. In trust-based systems (also known as reputation systems), cooperation is evaluated by past experiences, mapped on some numerical scale that represents the confidence between the participants. If a shared task attains its expected result, then increasing this value reinforces trust. Otherwise, trust is broken and future cooperation will be more carefully evaluated. The trust-based approach is typically implemented by a

reputation system, which supports the dissemination of reputation information between peers. On an incentive/trade based system, executing a shared task is rewarded with an amount of digital cash. Digital cash is used to establish an economic model among devices which implicitly evaluates device's past willingness to cooperate. The combination of both approaches, trust and incentive, forming a hybrid system, can be explored in order to obtain the benefits of both. Using a reputation system will allow users to evaluate the experience and incentive enables the creation of a resource market. This section discusses research results on these three approaches, aiming to support the development of a hybrid system for C3.

An extensive survey on popular reputation systems can be found in (Hendrikx et al., 2015).

3.2.1 Trust Systems

Trust systems are commonly used by Peer-to-Peer (P2P) file sharing protocols to implement a distributed reputation system in order to balance the download/upload ratio among peers. P2P reputation systems range from central trusted entities (e.g. (Singh and Liu, 2003)) to local views between peers (e.g. (Anagnostakis and Greenwald, 2004; Jun and Ahamad, 2005; Kamvar et al., 2003)). Since P2P is used in an Internet context, no assumptions exist about off-line system availability, clustering or transient connectivity, neither the moderate use of resources (computing cycles and bandwidth) is a major concern.

In contrast with P2P, by making their resources available, users of C3 are relinquishing of part of their device's available energy, thus reducing the personal benefits of owning the device. Conversely, the C3 model is beneficial for free riders as it extends the lifetime of their devices. Therefore, while in many P2P systems free riding is acceptable and a simple tit-for-tat algorithm (Cohen, 2003) is sufficient to ensure cooperation, we believe that the losses in Mobile Collaborative Cloudless Computing Environment (MC3E) claim for a more judicious evaluation of user's participation.

Reputation systems are being applied in other contexts, for example on the MANETs network layer for determining selfish nodes (Miranda and Rodrigues, 2010; Hu and Burmester, 2009; Mahmoud and Shen, 2010). Most of these systems support the unavailability of a central trusted entity by assuming a closed network model that rapidly disseminates selfishness information among peers. However, in an environment where device neighbourhood is expected to be highly transient, and clusters of nodes easily created, the migration of selfish nodes from a cluster to another would allow their reputation to be reset to a favourable or neutral status. Another example is BOINC (Anderson, 2004), a distributed system for public resource computing and storage.

On most of the existing reputation systems, users can discard their (bad) reputation by presenting themselves to the system with multiple identities. To solve this issue, an unchangeable ID could be assigned for example by a trusted third party that uses some off-line mechanism to enforce that each user does not acquire more than one pseudonym (Resnick et al., 2000). However such unchangeable ID raises privacy issues as it endangers users anonymity expectations.

From the studied alternatives, most of the trust systems are only applicable to situations where the shared resources have little to no value for the users, and where only the detection of selfish devices is an acceptable form of incentive. In scenarios where resources are valuable, users are unwilling to share them for free and a trade has to be established. This trade enables the needed source of incentive for users to accept the system.

3.2.2 Incentive Systems

In incentive systems, users receive a reward for sharing their resources. In Nuglets (Buttyan and Hubaux, 2001), a trade-based system for MANETs, authors define a currency in order to provide incentive for cooperation in packet forwarding. The amount of currency (Nuglets) owned by each device is stored in a trusted platform

module (TPM), a tamper resistant piece of hardware commonly used for storing sensitive data. Unfortunately, the dedicated hardware, reduces the number of possible applications to the few scenarios where all devices have such hardware available.

An alernative to Nuglets is the usage of a generic digital cash, such as Bitcoins (Nakamoto, 2008), a peer-to-peer electronic cash currency that removes the need of a central trusted entity by using transaction blocks distributed among all peers. Transactions in Bitcoin are represented by the exchange of virtual currency. To support the business model, brokers exist that convert the physical cash to electronic cash, and vice versa. However, the system is orthogonal to fraud, leaving to the users the responsibility to detect and solve any incident.

The work described in (Mahmoud and Shen, 2011) acknowledges the need to implement strong fraud detection mechanisms. To mitigate the impact of public-key mechanisms on the performance and lifetime of mobile devices, both propose to use hash functions once authentication between the devices has been established. This is an approach that can be equally followed in a hybrid trust and incentive system although the latter combines the trustworthiness of the virtual currency with a reputation mechanism.

Another requirement to be addressed in ad hoc environments is energy. This was addresses in (Luttenberger and Peters, 2011), describing an incentive system focused on being energy efficient and useful for this kind of environment where no central trustable entity exists. This work proposes a currency for exchange of trust without the use of public key cryptography. Contributing for an efficient CPU usage on battery dependent devices.

In (Iosifidis et al., 2014) the authors propose an incentive system for devices that share internet connectivity between them using WiFi direct, a WiFi communication where one of the peer devices assumes the role of an AP. Authors use the concept of Nash Barganing Solution (NBS) (Nash, 1950) to characterize the efficient and fair contribution of user resources and service allocation. NBS allows a decentralized solution that uses a virtual currency to pay users for the services they provide.

The main issue with incentive only systems, is that the collaborative nature of a system can still be exploited by unwilling users, to solve this issue and overcome the weaknesses of both trust and incentive systems, hybrid solutions were researched.

3.2.3 Hybrid Systems

One of the few systems that combines trust with incentive was developed for P2P systems (Fernandes et al., 2004). Authors apply a number of uncommon techniques in order to make it unattractive to exploit the system. This is achieved by rewarding devices each time an information is published. However, the incentive is delayed and not provided immediately. Additionally, at the retrieval of information, devices need to provide incentive to peers. To determine the participation of devices on the system, an honesty metric was established. When a participant is determined as being dishonest then no incentive is provided to that participant. If the system determines that a user is honest, the system will provide incentive for each published information about an interaction with another user.

More recently in (Bogliolo et al., 2012), a hybrid virtual currency and reputation system for a user centric network (where users create wireless communities, providing broader connectivity) was proposed. The model presented differs from the C3 as it includes the possibility of nodes to act also as relays of service requests and connectivity. Intermediate nodes that act as service relays, also enter the negotiated terms, this includes the first hop acts as a provider and establishes the cost of the required service from the source to the consumer device. A service provider can be for example a device that shares its Internet connection. The system addresses some challenges similar to the ones raised on the C3, however assumes that devices can reserve numerous resources for such a system, dismissing any optimization strategies to reduce resource usage.

The existing limitations on local storage of mobile devices leads to research on optimization techniques for dissemination of the reputation information. One of the

multiple directions of research is to predict contacts between devices, by researching human mobility, in order to reduce the requirements on needed storage space of each device.

3.3 Contact Prediction

Applications of research on human mobility for mobile computing have been mostly evolving around the opportunities for data dissemination and opportunistic routing. Common metrics for the characterization of human mobility are: $i)$ Inter-contact time (ICT), measuring the time between two consecutive contacts of two devices; $ii)$ Jump size, indicating the distance between two points where the device has stopped; and $iii)$ Pause time, that represents the time spent in the same place (Kim et al., 2006; Lee et al., 2009; Song et al., 2010a; Karamshuk et al., 2011).

The work described in (Chaintreau et al., 2007), reports a study on the ICTs of two distinct datasets. One is based on records collected from the access logs of WiFi networks. The second, named direct contact, contains records captured directly by devices, either produced specifically to be carried by users, or by exploiting the Bluetooth connectivity of off-the-shelf devices. Authors observed that the distribution of inter-contact times follows a power law for ICTs smaller than 1 day with the remaining presenting an exponential decay. Another research that supports the same findings can be found in (Karagiannis et al., 2010)). To improve the human mobility characterization, the authors of Hagle (Su et al., 2007) also consider the contact duration.

(Pietiläinen and Diot, 2012) goes beyond ICTs and addresses temporal communities (clusters of devices that are in range for a given time) and their relations. Authors extracted temporal communities from four distinct datasets, the largest of which considering the observation of 97 nodes over 9 months. To improve the study, authors obtained the social relations between users in some datasets, either by knowing the affiliation (on conference related traces) or by Facebook friendship graphs. This knowledge was used to establish social communities among users. In spite of the small scale

and duration of the study, authors presented two interesting conclusions. On one side, that the establishment of social communities has direct implications on temporal communities. On the other, that one particular class of devices, those with a high contact rate that are rarely seen in temporal communities, contribute significantly for the efficient content dissemination in opportunistic social networks.

Social communities are equally the focus of SocialCast (Costa et al., 2008). This work exploits the knowledge that humans tend to share interests and locations to develop an efficient routing protocol for publish-subscribe on Delay-Tolerant Networks. SocialCast uses Kalman filters for forecasting future contacts, based on previous observations of co-location between publisher and subscribers.

An innovative approach at predicting contacts, is presented in (Orlinski and Filer, 2013), where authors added a duration variable to communities detection, thus creating spatio-temporal communities. The community relevance is increased proportionally to its duration, which improves the efficiency of cluster based data delivery in Pocket Switched Networks (networks formed by encounters between devices carried by humans). It was shown that spatio-temporal communities can contribute to improve the efficiency of information dissemination in these opportunistic networks. Simulation experiments were conducted in the same datasets used by the Haggle project.

Contact prediction has also led to the development of routing algorithms. These constrain the number of retransmissions, in order to reduce network congestion due to spurious flooding. Prophet (Lindgren et al., 2003) uses this strategy on MANETs, by restricting message dissemination to devices that have a higher probability of contacting the destination. Prophet uses an history of previous events in order to predict future encounters by calculating the probability of a device being useful in a nearby future for packet routing.

In (Huang et al., 2015) authors propose PreKR, a framework that optimizes the forwarding on opportunistic networks by using a kernel regression based estimation for link pattern prediction. Using historical observations of network maps on three

datasets (one of them being the dataset of the Haggle project), PreKR determines the probability of a recurrence of a link between two devices. Authors determined that PreKR outperforms all other prediction methods, including Prophet. The distinguishing factor was the use of kernel regression, that allowed PreKR to achieve an accuracy of more than 90%.

Bubble Rap (Hui et al., 2011), is a socially influenced routing protocol, that leveraged on the mobility traces of the Haggle project to infer temporal communities. Authors used the K-Clique (Palla et al., 2005) and weighted network analysis (WNA) algorithms (Newman, 2004), two forms of centralized algorithms, to extract communities from mobile traces. The two algorithms were chosen for their features. K-Clique detects overlapping communities, but requires as a complex configuration process before being used. WNA is easier to set up, but is incapable of detecting overlapping communities. These algorithms are specially useful for forwarding applications, where a path needs to be predicted. However, for applications where a prediction of temporal communities is enough, K-Clique and WNA proved to be highly complex.

In (Song et al., 2010b), authors evaluated the limits of predictability of human mobility by analysing the movement patterns of mobile phone users and found that the observed mobility was highly predictable, where most users are localized in a finite neighbourhood. Authors estimated that there is a potential 93% average predictability in user mobility.

In (Foell et al., 2014), authors also predict human mobility, but limit it to a user being present on a bus/bus stop. Authors detail a number of possible prediction algorithms for this scenario and were able to predict future bus stops using historical data. These predictions would also suit the C3 environment, if we consider that all bus riders are in range and participate on the C3.

The study of the related work suggests that research on contact prediction is still at an embryonary stage. Only a few of the works address the problem of estimating the moment of which future contact will happen, as well as estimation of the number of devices in transmission range. In contrast, research has been focused on the prediction

of links between two devices across a MANET.

3.4 Mobility Patterns & Models

The research on dissemination algorithms using contact prediction benefits of mobility models that purport the features previously observed in human mobility. Such models can be expanded and configured to determine and observe the evolution of human mobility, for example when more devices, possibly with different characteristics, are added. To create this statistical model, researchers use a dataset containing human mobility and extract a number of metrics that are later studied in order to obtain a statistical model. Mobility models can be split into two categories, according with this requirement: $i)$ Pure synthetic mobility models; and $ii)$ Trace-based synthetic models.

Pure synthetic mobility models employ random distributions to simulate device movement. A classical example is the Random Waypoint Mobility Model (RWP) whose simplicity in the generation of mobility scenarios facilitated cross comparison of mobile applications and protocols, see (Johnson and Maltz, 1996; Perkins et al., 2003) for examples. However, it has been shown that, in addition to their disparate modelling of human behaviour, synthetic mobility models typically bias node distribution in a non-natural way (Bettstetter et al., 2003). Limitations of the RWP have been addressed, for example in (Gyarmati et al., 2008), were a variation of the traditional RWP to produce patterns presenting the same inhomogeneity as found in human mobility was proposed.

Trace-based synthetic mobility models, on the other hand, attempt to mirror patterns of human movement by modelling nodes behaviour according to the same probabilistic distribution functions observed in traces. The mechanisms used for collecting data inspiring trace-based mobility models can be arranged in two categories. Intrusive approaches (for example (Piorkowski et al., 2009; Thiagarajan et al., 2011)) are those that obtain their data directly from the device carried by the user. These approaches benefit from the precision of the data, captured by dedicated software or

hardware. Unfortunately, these studies are constrained by the considerable amount of resources involved, which limit their time scale and number of participants and may bias conclusions concerning the identification of patterns.

Non intrusive approaches use logs collected by external devices (like access points or indoor-localisation devices) to produce traces with the user location at each instant. In spite of the privacy issues raised with the collection of the data, non intrusive approaches are those that present the capability to scale more in both number of users and time span. Unfortunately, surveys on mobility models (Aschenbruck et al., 2011; Karamshuk et al., 2011) suggest a scarcity of data from mid-2008 onward, thus excluding the massification of mobile devices observed with the emergence of the last generation of smartphones and tablets. If available, more recent traces could evidence the emergence of new mobility and contact patterns among users.

This is the case of (Tang and Baker, 2000) which reports on the network traffic and user mobility in an University wireless network during the 12 weeks of the winter semester of 1999/2000. Unfortunately, in early 2000, wireless networks were still at an embryonic stage[5] and its conclusions are expected to be invalidated by the considerable increase of the number of users. In addition, evolution in mobility can only be observed in a larger time scale.

The WiFi network of the Dartmouth College has been serving for collecting a considerable number of traces, for example during the 17 weeks of the 2003/2004 winter semester (Henderson et al., 2008). Authors used the logs to model real user tracks and defined a threshold walking speed, below which users were assumed to have stopped before moving to the destination. This knowledge was used to define a trace-based synthetic mobility model (Kim et al., 2006) inspired on the mobility patterns of 198 VoIP handsets. The model addressed social, spatial and temporal features and considered hotspots, workday/weekend distinction, and mobile and stationary sets. In comparison with the work presented in Chap. 6 of this thesis, the study of 2003/2004 evaluates a larger number of access points, but a lower number of users and a shorter

[5]First IEEE802.11 standard version was published in 1997.

time frame. Unfortunately, its age prevents it from considering a number of recent advances, like the most recent wave of mobile devices, such as the iPhone and Android OS based smartphones (debuted respectively in 2007[6] and 2008[7]).

Results of a two month study on the eduroam infrastructure of the universities of Minho and Vigo can be found in (Mulhanga et al., 2011). Authors found that the APs with more users are not necessarily the ones with more network traffic. In addition, the paper evidences a weekly use pattern for this network, with the vast majority of users connecting only on weekdays. In terms of mobility, authors conclude that 90% of the users connect to more than one AP monthly, with about 35% visiting at least 5 APs. The study followed an interesting methodology, for example by associating access points to physical spaces, thus allowing to separate network traffic originating in residential from academic areas. Unfortunately, the small analysis period of this study makes the notion of mobility disperse in time and of little relevance in the characterization of real mobility.

3.5 Summary

Architectures that address mobile application requirements by using cloud execution can be arranged according to the location of the resources they make available. The traditional cloud, a cloud that is supported by a nearby device, or a form of cloud that uses mobile devices as cloud nodes which we call the Collaborative Cloudless Computing (C3).

The most prominent research approaching the C3 system model ignores the trust and incentive issues raised by an architecture of this kind. To develop a framework for C3 that addresses this requirement two approaches are found in the literature. A reputation system that enables trust among participants or a digital cash protocol that provides a form of currency exchange. However, there are other challenges amplified

[6]See "Apple iPhone" at: http://en.wikipedia.org/wiki/IPhone (Last access on: 19 April 2015)

[7]See "Android OS" at: http://en.wikipedia.org/wiki/Android_(operating_system) (Last access on: 19 April 2015)

by the limitations of the environment and therefore, claiming for a moderate use of the network and local resources due to the limited computing power, available energy and storage capacity. The optimization of the dissemination algorithm of the trust and incentive system is a step in the right direction by contributing to attenuate many of the problems.

Contact prediction is used on literature for optimization of not only dissemination algorithms, but also for link prediction on opportunistic networks, path anticipation on routing protocols and other mobile applications or network protocols. Most of the contact prediction algorithms use small datasets or the output of synthetic mobility models to extract metrics that are supposed to portrait human mobility features. However, these metrics dismiss the trust and incentive system requirements of the C3. In order to improve this, new metrics and new datasets need to be investigated.

4

Mobile Collaborative Cloudless Computing Environment

The Collaborative Cloud Computing (C3) paradigm is our vision where the concepts of cloud computing and mobile devices are merged. As opposed to Mobile Cloud Computing that connects to a traditional Cloud or nearby Cloudlets, in C3, the cloud nodes are the mobile devices themselves. We propose an architecture for a C3 framework named Mobile Collaborative Cloudless Computing Environment (MC3E), that it's expected to use off-the-self devices and operating systems.

4.1 Framework Architecture

The deployment of Mobile Collaborative Cloud Computing Environment (MC3E) is based on the execution of a component in all participating devices. When enabled by the user, this component will work reactively, answering requests from applications running locally or on other devices. This framework fits within the middleware class, placed between applications and the network, which aims to abstract the applications from the complexity inherent to the use of the C3.

The MC3E framework is responsible for supporting the delegation of application defined tasks. To solve some of the challenges, auxiliary mechanisms will be in place, although the application is abstracted from their existence.

The structure of the MC3E framework is depicted in Fig. 4.1. The platform is inter-

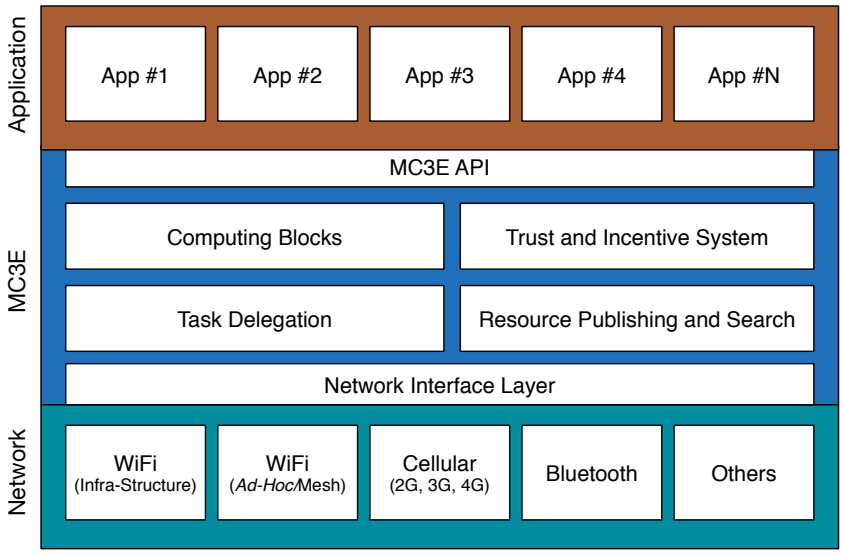


Figure 4.1: Mobile Collaborative Cloud Computing

nally split into six major modules, described below.

4.1.1 Computing Blocks

Computing blocks are defined as the basic units of work in MC3E. Computing blocks abstract several types of resources or activities, for example:

- Computing, a function that can be simple as a discrete Fourier transform or a more complex speech recognition function.
- Specific hardware capabilities, like GPS.
- Shared memory or caches, allowing other blocks to store the results of a task, making them available for other devices.
- Task partitioning, splitting a task into simpler ones, represented by other computing blocks, possibly to be executed in parallel allowing tasks to complete faster.

A computing block could be available on all devices or only on some, following some replication policy, which can be determined by the frequency of use, or by features on the device. For example, one block could be only available on some devices

with Discrete Cosine Transform (DCT) hardware acceleration. A common ID infrastructure should be used to publish available computing blocks, using for example, the Structure of Managed Information (SMI) supported by IANA (Internet Assigned Numbers Authority) (McCloghrie et al., 1999). Invocations to a computing block are similar to a traditional Remote Procedure Call (RPC), with the caller sending the computing block ID and the arguments and receiving the return value.

Software computing blocks are downloaded in advance by the user from some trusted site, like the device manufacturer or OS distributor website, or even from an application web store. By avoiding code transfer between devices, MC3E contributes to reduce the cost of task distribution and to increase the sense of privacy, given that no untrusted code will be executed in any device. The execution of each block has an associated computational cost, which is used to distribute load and reward devices. The computing block is also responsible for ensuring data isolation, between the task and the personal data on the host device.

Finding an adequate level of complexity for computing blocks is challenging given that no previous research exists on automatic decomposition of generic computing tasks. Research directions to be pursued include the use a programming language that eases the use of parallelism. StreamIT (Thies et al., 2002) is a programming language and compilation infrastructure for stream programs, which eases the use of parallelism. Given a stream graph, with computation and communication resource needs, StreamIT finds the schedule of execution that optimizes the resource usage using parallelism. Another line of research to be investigated is the use of domain specific languages, where the programmer is unaware of the underlying parallel computing architecture, but provides hints on how to better support this decomposition of computation to reach simpler parallel computing blocks (Chafi et al., 2010; Garland et al., 2008), an approach that has been experimented for transferring computing blocks to Graphics Processor Units (GPUs).

4.1.2 Task Delegation

The task delegation module is responsible for accepting, coordinating and answering to task requests from the computing blocks and from other nodes. This module is also responsible for ensuring the reliability of the computing block results of other devices, as well as ensuring the data privacy.

Task delegation relies on a mechanism that compares the costs of local execution, delegation on a cloud in the wired infrastructure and task delegation to other devices (Kumar and Lu, 2010). If the late shows to be advantageous, the module uses a scheduling algorithm to ensure a fair distribution between nodes, which implies awareness of the resources available in other users' devices to avoid degrading user experience on their own devices.

As a last resort, results reliability can be evaluated from user feedback. Alternatively the results can be compared with the execution of the same task locally or on another device. However, this approach doubles the computational effort demanded to the C3 and is vulnerable to collusion. In spite of the lack of efficiency of both approaches optimistic solutions, relying on device's reputation should be the rule, rather than the exception.

4.1.3 Resource Publishing and Searching

The resource publishing and search module advertises the computing blocks available on the device. At the same time it interacts with the reputation system, to verify the reputation of the answering nodes. Finally, the module also maintains neighbourhood activity statistics to help select devices better suited to perform a task. Historical values of signal strength and quality and battery, will influence the choice of devices/resources.

Resources could be located pro-actively or reactively. Proactively, a resource mapping is permanently kept by each node. A reactive approach implies that the resource

mapping is done on demand during task delegation. None of the solutions is perfect: Proactive, for wasting bandwidth and energy with potentially irrelevant information; reactive, by imposing a non-negligible delay during the task delegation.

Resource publishing and search may use different communication paradigms, such as publish/subscribe, distributed hash tables (DHT) or flooding, with the most appropriate type of communication being dependent of the characteristics of devices, network and the C3 itself. In flooding (Cheng, 2002) messages are delivered to all participants. In the scope of resource location, messages can be queries (reactive) or resource voluntary advertisements (proactive). Flooding reduces the middleware complexity and improves battery consumption in the scenarios where all nodes are within transmission range. Publish/subscribe is an asynchronous communication model where subscribers are notified of events by publishers (Eugster et al., 2003). The content-based publish/subscribe model allows subscribers to specify filters for events that they wish to receive, as opposed to topic-based publish/subscribe where the subscribers use pre-defined topics for subscription, limiting subscribing options. Distributed hash tables (DHT) provide a data structure were the content, a (key, value) pair, is dispersed on participating nodes. The search on a DHT is related to its structure. One of the most common is a circular structure where the content and nodes use the same space for IDs (e.g. Chord (Stoica et al., 2001)). Other alternatives exist for resource location (Meshkova et al., 2008), such as a XOR-tree (Maymounkov and Mazières, 2002), Pastry (Rowstron and Druschel, 2001) or CAN (Ratnasamy et al., 2001). The volatile nature of the underling network, where neighbouring nodes could change frequently, has strong implications on the performance of DHTs in mobile scenarios.

4.1.4 Trust and Incentive System

The trust and incentive system module answers requests from other modules about the trustworthiness of MC3E participants. This information is used by the resource publishing and search module, to estimate the reliability expected to the requests that the node makes. It is also used in the process of deciding if a request should be ac-

cepted, and to account the computational effort required from each device.

A trust system is needed to mitigate selfish nodes impact on the cooperative cloud. In the scope of this work, a selfish node is a node that repeatedly ignores MC3E task request but uses other nodes to deploy its tasks. At the network layer level, selfish nodes are those that ignore packet relay requests use other cooperative nodes to relay their packets (Hendrikx et al., 2015). Any of these situations negatively impact the cloud fairness and performance.

Reputation systems can be vaguely defined as systems that allow users to give some form of credit to another after a transaction between both. Implementations of this definition can be, for example, in the form of trust information that is kept by the client and advertised to third parties.

We will be using some principles of other reputation systems on ad hoc environments to implement a similar system on the MC3E framework, for example considering the availability of a central trustable store for trust information or the transitivity characteristics of the neighbourhood. This trustable source is defined as a secure entity for managing the reputation information of each node.

A reputation system is needed in a scenario where users require motivation to trust other devices and where users have may have different goals. For the dismissal of the reputation system we can assume two options: the users share the same objective/task, or the users built a closed trust group and we can consider the C3 a private cloud.

Users are expected to have multiple virtual IDs, using a form of pseudonym, in order to ensure anonymity of a users' identity. This model was investigated in (Miranda and Rodrigues, 2006) where anonymity was combined with reputation, balancing both, using well known cryptographic solution: a public key infrastructure (PKI) along with blind signatures to provide anonymity. However, the impact of using a PKI on the C3 has to be researched.

Another alternative to the creation of trust in the reputation system is using an incentive system. An incentive system is inspired on virtual cash to provide a currency

used for trading between devices, effectively creating a virtual marketplace for the C3.

4.1.5 API

The MAPI (MC3E API) is responsible for exposing the MC3E platform functionality to the application. One of its roles is to hide from the application programmer the inherent complexity of the MC3E utilization by abstracting it in a few simple operations. The MAPI routes task requests to local computing blocks or to the task delegation module. The MAPI also exposes MC3E's control and status functionalities, such as authentication and access control, reputation system status, resource map, and current usage.

Defining an API is challenging given that most of the standard Cloud APIs are focused on IaaS clouds (Denman, 2010; Metsch, 2010). PaaS Clouds (which provide a platform) such as MC3E still require standardization, with current APIs being defined by each provider.

4.1.6 Network Layer Interface

The network layer interface is responsible for the abstraction of the different network technologies, presenting a single interface to the other modules.

This module is responsible for providing user data privacy functionalities during the transport, using for example TLS as a transport protocol (Dierks, 2008), and for the adaptation of MC3E modules to the network technology in use.

4.2 Vertical Issues

Some problems that need to be addressed for a successful implementation of the MC3E are common to multiple modules. These problems are discussed in the next sections.

4.2.1 Security

Security is a global goal in our architecture. To safeguard user data privacy, and achieve a more secure platform, we have to consider it on all modules that deal with private user information or that need to ensure data integrity.

Any secure system will ensure five different aspects, integrity, authentication, confidentiality, access control and privacy. A public key infra-structure (PKI) (Adams and Lloyd, 2002) such as OpenSSL will address the first two items (integrity and authentication) (Stallings, 2010).During task input and output, data transmission can be secured by TLS (Dierks, 2008), using the same PKI, to ensure that no Man-in-the-Middle attack is done to the communication. Man-in-the-Middle attacks are done forging a server certificate and intercepting the client connection, presenting the forged certificate to the client and routing communications trough the attacker. Although, PKI assumes a central trusted entity this should not be considered as a problem given that our model assumes a periodic connection to the Internet. TLS and PKI introduce an overhead in communications and alternatives have to be studied to evaluate if a PKI is the best solution, or if solutions like those discussed for reputation system are preferable.

Ensuring user data privacy when executing a task on a foreign device is challenging. An approach to be investigated involves a judicious distribution of the data by several participants, to ensure that information that allows inferring the global context and content is never delivered to a single node (for example, randomly distributing words by the participants in a text translation task). However, this approach fails to address the scenarios where the number of participating devices is small and some compromise solutions will have to be devised.

4.2.2 Battery Usage

Of the different components available on a mobile device, the wireless interface and the CPU are those with a bigger on battery consumption (Feeney, 2001). Current research shows that the most battery consuming wireless interface during data

transfers is the cellular interface, while Wireless Ethernet or Bluetooth are more conservative (Perrucci et al., 2009; Balani, 2007). However, battery usage could be further optimized using traffic modelling techniques. Research shows that sending traffic in bursts allows wireless interfaces to enter the power saving mode more often (Hoque et al., 2011). Also, on a MC3E, it is expected that nodes are in proximity, which can provide additional savings, given that less transmission power will be needed and an higher throughput can be achieved (Electro-Mechanics, 2010).

To further optimize battery usage, multiple wireless interfaces could be used, with vertical hand-off occurring when higher throughput is needed (Pering et al., 2006; Ananthanarayanan and Stoica, 2009). For example, one could use Bluetooth for neighbourhood discovery and Wireless Ethernet for task deployment and result collection. Data transmission by Bluetooth is expected to have six to ten times lower power consumption than the same data transmission using Wireless Ethernet, assuming the more conservative transmission mode of both technologies (Balani, 2007; Electro-Mechanics, 2010).

Previous knowledge on battery charging profile (Ferreira et al., 2011) allows the MC3E to influence decisions on task deployment. This allows delay tolerant applications (Burleigh et al., 2003) to be run on times where the device is charging, minimizing impact on user experience, and also providing a somewhat large number of collaborating devices.

The combination of multiple techniques should allow the energy consumption to be lower than the one obtained when using the traditional cloud, or at least match it.

4.3 Use Case

As an example on the use of the MC3E framework, the use case presented in the motivation is revisited:

The automatic translation application used by Alice solicits a new computation to MC3E.

The arguments passed in the invocation are the audio translation computing block identifier and the captured digitized audio. When MC3E receives the arguments, it will use the audio translation computing block to initially process the data. This allows MC3E to ensure user data privacy by partitioning the data for the parallel execution of the task. Afterwards the task delegation module requests the resource publishing and search module the identity of devices that provide the computing blocks required for audio translation. The list of available devices is delivered to the reputation system module that returns a subset of devices that believes more appropriate/trustable. The task delegation module then distributes audio clips to different devices while keeping some locally. The result of a computing task returned by a computing block could be the final result or an intermediate result that includes a list of computing blocks and a new set of arguments. If the results are intermediate, the MC3E platform is responsible for triggering a new resource search and delivery of arguments to the new blocks. At the end of the task execution, the reputation system rewards devices whose results were considered correct.

4.4 Summary

This chapter presented a framework named Mobile Collaborative Cloudless Computing Environment (MC3E) that uses idle CPU cycles of devices in the neighbourhood to provide the computing power required by mobile applications running on a mobile device and which is currently provided by resourceful servers. This is an appealing concept, which can contribute to alleviate the load on wireless cellular networks, reduce application response time and improve availability. However, the implementation of a MC3E raises a number of non-trivial challenges. These challenges appear mostly from the ad hoc nature of the networking environment, characterized by the lack of trust among the participants, insufficient motivation for cooperation, unstable connectivity, devices limited resources and latency.

5

Hybrid Trust and Trade System

The participation on Collaborative Cloudless Computing (C3) consumes battery, memory and CPU time of the mobile devices. It is assumed that users value their resources and therefore prefer to trade instead of donating them. This is in contrast with what has been observed in peer-to-peer file sharing, where resources (mostly bandwidth) have no value to users. Therefore, C3 would benefit of a services market that fosters collaboration by allowing users to "sell" resources when they are not particularly keen on their device usage. However, the implementation of a services market requires a long-term memory that allows users to collect their rewards days or weeks later, possibly from a distinct set of devices.

The services market has similarities with other on-line markets, of which eBay or AliExpress are good examples. Both are characterised by the distrust between participants and by the difficulty in penalise misbehaviours. EBay addresses misbehaviour with a central reputation database that stores transactions rating feedback provided by the participants and leaves to the user the final decision of performing transactions with any other participant. However, a central reputation database would invalidate the off-line and distributed nature of the C3 by increasing the load on the wireless or cellular infrastructure due to the additional traffic. Multiple alternatives have been proposed ranging from incentive to reputation systems (see (Hendrikx et al., 2015) for a survey). Unfortunately, most are targeted to a particular networking environment and ignore the transitivity of the connections, assuming either that punishment/rewards mechanisms are not applicable outside the network scope where the interactions are

taking place (e.g. (Felix Marmol and Perez, 2010)) or that a trusted authority storing the reputation is permanently available (e.g. (Houser and Wooders, 2006)). The trust and incentive system proposed for the Mobile Collaborative Cloudless Computing Environment (MC3E) provides a hybrid reputation/digital cash service for leveraging mobile collaborative computing scenarios with a long term memory, called the Hybrid Trust and Trade System (HTnT). The approach considers a highly transient neighbourhood, where nodes have occasional Internet connectivity and therefore can be applied to a broad range of mobile applications. HTnT is particularly keen on privacy preservation, enforcing the use of pseudonyms on all the interactions between participants of the C3.

5.1 Requirements

A successful deployment of a Mobile Collaborative Cloudless Computing Environment (MC3E) depends of the user acceptance. The motivation will come from the user perception that the model is inherently safe and fair. The following requirements to achieve this goal have been identified:

R1: Fair and proportional cooperation reward

Mobile Collaborative Cloudless Computing Environment (MC3E) should be fair, offering a reward proportional to each user contribution. This can be achieved by associating a numerical value to the resources made available and used. A currency allows a fair trade system, where the sharing user establishes an amount of credits to receive (a price) for his effort in executing the task. Credits can be used later for acquiring services from the Mobile Collaborative Cloudless Computing Environment (MC3E).

R2: Discourage fraud

HTnT must discourage, detect and prevent any attempt to subvert the reward model, in particular, to reward users for services that they did not provide. Although a reliable monetary system addressing R1 should be sufficient to inhibit selfish users, there are approaches that can be followed by misbehaving users to circumvent a fair trade and

which can only be addressed with reputation information. In addition, fraud detection should equally address social misbehaviour, such as damaging users' good reputation.

R3: Decentralization

The service should be able to operate without permanent access to a central trusted entity. HTnT must assume that devices connect to the Internet occasionally, at convenient moments like when the mobile device is charging its batteries. These connections are not expected to occur while the devices are using the MC3E and, even if they occur, will penalize the latency of the service execution.

R4: Platform independence

Mobile devices exhibit a wide range of combinations of hardware and software. MC3E aims at being platform independent and therefore must operate under a minimal set of requirements, dismissing the use of any kind of specialised hardware, of which tamper-proof devices are a good example.

R5: Respect user's Privacy

Users of the MC3E must benefit of their right to privacy, by not having any of their personal information disclosed. Permitting each user to hide his identity using a pseudonym can facilitate anonymity. Pseudonyms should be changed frequently, given that records of an anonymous user's typical usage pattern or location can be associated to disclose his real identity. Interestingly, anonymity conflicts with the need to maintain user reputation as it can be used by malicious users to obtain a clean reputation record, a process known as white-washing.

R6: Efficiency

To cope with its operating environment, the system must make a moderate use of the device's resources. Efficiency is attainable by using algorithms that minimise computations and by a moderate use of the network and storage.

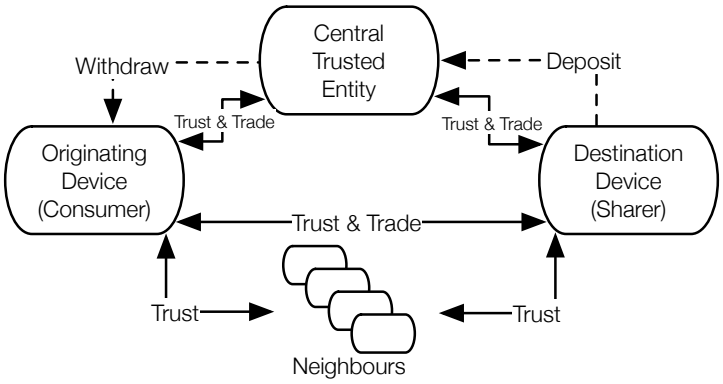


Figure 5.1: Relationships among different parties

5.2 A Hybrid Trust and Trade System

The hybrid trust and trade system for the MC3E combines a reputation system, to identify malicious nodes, with an incentive system, to reward the devices that share their resources. One instance of the algorithm is executed for each transaction. It should be noted that the algorithm is orthogonal to the C3. Therefore, particular aspects like the initial amount of virtual currency provided to each user or the cost of each transaction are left as configuration options.

5.2.1 Overview

The system identifies four participants in a transaction, depicted in Fig. 5.1. The central trusted entity (*CTE*) issues and validates virtual currency and manages user's currency accounts and reputation. CTE has no intervention at transaction time; the *consumer* is the user who starts a transaction by expressing his intent of using services made available by another device; the *sharer* is the user that accepted to perform the services; *neighbours* are devices in proximity of the consumer and the sharer. The presence or participation of neighbours in a transaction is not required.

The interactions between the devices and the CTE are orthogonal to individual transactions. They are expected to occur at the user's convenience (for example, during device charge cycles). On each interaction, devices *Deposit* and *Withdraw* virtual

currency and learn the most up-to-date reputation information of some other participants. Deposits add to the account the amounts collected by providing services to third parties. Withdraws transfer virtual currency to the device' digital purse, allowing to acquire services from the peers.

The CTE equally serves as a reputation keeper and advertiser. The reputation of the users is disseminated by the CTE when devices perform deposit and withdraw operations and among neighbouring devices.

To preserve anonymity, users have two identities. A user's immutable ID is only known by the CTE. Pseudonyms are used in interactions with the peers, these are generated and digitally signed by the central trusted entity. Users are expected to frequently change their pseudonym, in cooperation with the CTE. To improve user privacy, the pseudonym can be hidden from the CTE using for example the process described in (Miranda and Rodrigues, 2006).

5.2.2 Virtual Currency

Virtual currency will numerically represent the amount of work serviced to MC3E by each user. In the scope of this work, the relevant proprieties expected to be exhibited by a virtual cash protocol are: $i)$ *Uniqueness* for the detection of forged currency, $ii)$ *Privacy*, preventing attempts to associate user identity to past transactions, $iii)$ *Transferability* of the currency between devices, and $iv)$ *Off-line Validity*. The *digital cash protocol number 4* (Schneier, 1995) is an example of virtual currency protocol satisfying these properties.

Each monetary unit is a data structure with fields for its value and uniqueness and ID strings. The structure is digitally signed by the CTE. Each element of the ID string is a pair of bits, usually designated as *left* and *right*.

In our system, the digital cash protocol is extended with the concept of *TwinCoin*. A TwinCoin (depicted in Fig. 5.2) is an aggregation of two coins, each exhibiting, alone, the same properties of the original coin. The two coins are associated by sharing the

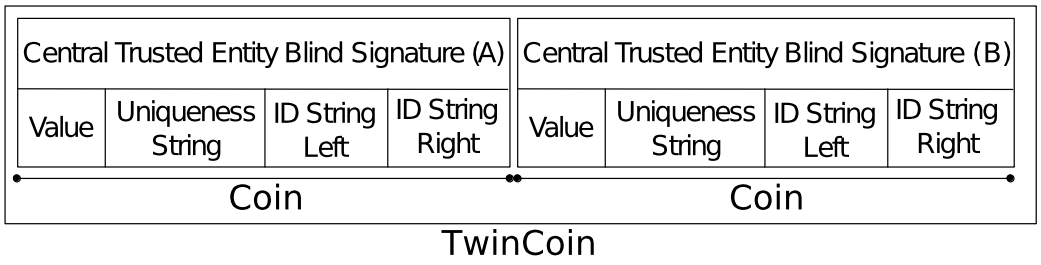


Figure 5.2: TwinCoin Format

same value, uniqueness and ID strings but with different digital signatures from the CTE.

A single coin is valueless without its corresponding counterpart. Conversely, a TwinCoin can only be obtained from the reunion of the two originating coins. The value of a TwinCoin is converted by the CTE once it receives the two coins that compose it. TwinCoins can be used at most once. That is, devices may only use virtual currency that has been previously deposited on the CTE.

5.2.3 Transactions

Transactions occur in three steps, depicted in Fig. 5.3. In the first, the consumer announces to the neighbourhood the interest in delegating a task, and selects the sharer. The price settlement process is orthogonal to our protocol and can consider both technical and social aspects. Examples are the computational effort required, the reputation of the participants and the device's battery level. The consumer and sharer candidates are expected to identify themselves with the reputation certificates issued by the CTE and which are described below.

In the second step, the consumer assigns a TwinCoin from his purse for the transaction and delivers to the sharer one of the coins.

The final step is initiated when a proof that the service has been executed is accepted by the consumer. At this phase, the consumer is required to deliver the second coin of the TwinCoin to the sharer and to remove the TwinCoin from its purse so that

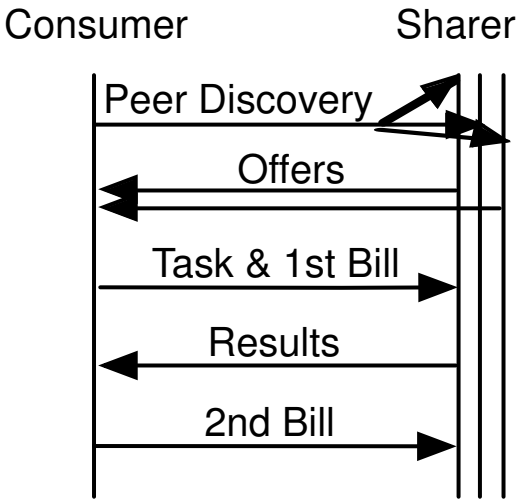


Figure 5.3: Task transaction

it cannot be reused. Proof of service completion will rely on the application for which the algorithm is being used and is orthogonal to our exposition.

5.2.4 Interactions With the CTE

The CTE keeps a virtual currency account for each user. To act as consumers, users must withdraw virtual currency from this account, converting it to TwinCoins to be stored at the device's electronic purse. Figure 5.4 presents the algorithm followed to create TwinCoins. The blind signature of the TwinCoins respects the original digital cash algorithm and is used to increase user anonymity. The system does not support the TwinCoin fractioning. It is up to the system and user to define TwinCoins with values adequate for the transactions that may be performed in the future. A user may have in its possession large amounts of TwinCoins, decreasing the need to contact the CTE between transactions.

To benefit from the virtual currency earned as a sharer, nodes must contact the CTE to deposit them. As depicted in Fig. 5.5, deposits are performed by the sharer, with the delivery of each TwinCoin to the CTE. In possession of a complete TwinCoin, the CTE can confirm its validity and credit its value in the sharer's account. The CTE takes a commission from each deposit that is later distributed to the accounts of devices with

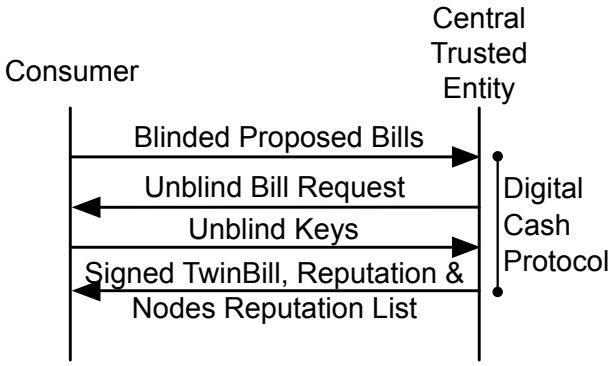


Figure 5.4: Withdraw operation

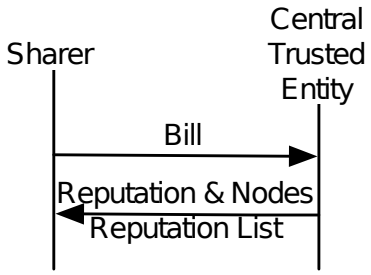


Figure 5.5: Deposit operation

a reputation above a certain threshold.

The third interaction type between devices and the CTE is the change of the pseudonym of the former. The real immutable identity of the users is only known by the CTE. In the interactions between devices, the users present themselves using a pseudonym, certified by the CTE. The pseudonym change is further addressed below in the section that discusses privacy.

5.2.5 Reputation Information

To discourage misbehaviour, the system complements virtual currency with reputation information. Reputation information aims to provide knowledge about users past behaviour, allowing users to make more informed choices on the devices with whom they will cooperate. Two classes of reputation are defined: *global reputation*, managed by the CTE, and short-term *local reputation*, built directly by the devices according to their experience on transactions happening in the intervals of their contacts

with the CTE.

5.2.5.1 Global Reputation

Reputation information for each user is managed by the Central Trusted Entity (CTE), and updated when devices contact the CTE. Reputation information for each user u, is a value $GR_u \in [-1.00, +1.00]$, with each successful or unsuccessful transaction respectively adding or subtracting from the value. The smoothing formula depicted in Eq. 5.1 is used to weight the result of more recent interactions with user's past reputation.

$$GR_u = \alpha \times GR_u + (1 - \alpha) \times R_t(C_t) \tag{5.1}$$

Where the reputation value for a transaction R_t depends of the overall cost agreed by the participants for the transaction (C_t) and of its outcome according to the function presented in Eq. 5.2

$$R_t(C_t) = S_t \times (1 - \frac{1}{C_t + 1}) \tag{5.2}$$

S_t is a success factor, that will be respectively $+1$ or -1 if the transaction completed successfully or not, t stands for transaction. Recall that the consumer delivers one coin to the sharer with the request for the execution of a task and the second upon confirming its successful completion. Therefore, for the CTE, a transaction will be considered successful if a TwinCoin is delivered by the sharer. On the contrary, a transaction is considered unsuccessful if only one coin is delivered by any of the participants. Users are expected to learn from their past experiences, and therefore, at most one unsuccessful transaction report is accepted by the CTE for each pair of pseudonyms.

The CTE relies on the devices to advertise theirs and other users' reputation. *Reputation certificates* are data structures digitally signed by the CTE containing the user pseudonym, GR_u and issue date. On every interaction between the CTE and the users,

the CTE delivers a number of reputation certificates to the device, facilitating reputation dissemination and reducing the number of contacts between the devices and the CTE. The list of reputation certificates delivered includes the one of the user contacting the CTE and a biased selection of reputation certificates of other users, according to a policy to be discussed bellow.

5.2.5.2 Local Trust

Local trust is built from the experiences of each device with its neighbours. This information can be shared locally, thus rapidly disseminating reputation information which may diverge from the one present in the reputation certificates.

5.2.5.3 Reputation Learning

A quest for reputation is expected to be initiated by any of the participants in a transaction, whenever the global information made available by the peer is considered insufficient. Possible criteria are the date if the reputation certificate or its low value. A node triggers the reputation learning procedure with a 1-hop broadcast request of reputation information to its neighbours. Neighbours carrying reputation information about the node reply with a point-to-point message. This information can either come from their local reputation database or from the list delivered by the CTE to the device. Reputation information is digitally signed and therefore cannot be forged, although it is vulnerable to collusion, as it will be discussed below.

The rationale beyond the decision to trust on a specific device is outside the scope of this work. We anticipate that the decision should weight the most up-to-date reputation certificate obtained (which can be provided either by the peer or by one of the neighbours) and recent local reputation provided by peers.

Table 5.1: Misbehaving actions considered

M1	The consumer repeatedly uses the same TwinCoins in different transactions;
M2	The sharer reuses TwinCoins obtained in the scope of a legitimate transaction;
M3	The consumer does not deliver the second coin;
M4	The sharer fails to deliver the task results to the consumer;
M5	The sharer falsely accuses the consumer of not delivering the second coin;
M6	The consumer falsely accuses the sharer of not delivering the results;
M7	A participant forges his reputation information;
M8	The consumer forges reputation information of other devices;
M9	Participants collude to degrade other user reputation;
M10	Participants collude to gain unfair benefits;

5.3 Discussion

Attempts to gain undeserved benefits by forging virtual currency, as stated in Requirement R2 are assumed to be resolved by the virtual currency protocol, whose proof of correctness can be found elsewhere (for example in (Schneier, 1995) for the *digital cash protocol number 4*). However, due to the distributed nature of the system, some forms of misbehaviour cannot be completely prevented. For these, the system makes its best to ensure that remaining users are alerted through the reputation system. Table 5.1 lists the misbehaviour actions considered. The procedures to mitigate their impact is discussed in the following paragraphs.

Misbehaviour actions M1 and M2 are resolved by the CTE, supported by the virtual currency protocol, which provides the mechanisms necessary to detect these actions and identify the author.

HTnT addresses misbehaviour actions M3–M6 in conjunction. Both the consumer and the sharer can accuse the other partner in a transaction by delivering to the CTE only one of the coins of a TwinCoin. The result of any of these actions is the application

Table 5.2: Impact of several misbehaving actions

Misbehaviour	M3	M4	M5	M6	None
Consumer	F: - R: ↓	F: ↓R: ↓	F: - R: ↓	F: - R: ↓	F: - R: -
Sharer	F: ↓R: ↓	F: - R: ↓	F:↓R: ↓	F: ↓R: ↓	F: - R: ↑

F = Financial impact
R = Reputation impact

by the CTE of a penalty to the reputation of both participants. In addition, TwinCoins can only be used once and therefore, become invalidated by either an accusation or by a successful transaction.

The rationale behind this approach is that if both parties complain, it may not be possible for the CTE to judge which of the participants effectively misbehaved. However, well behaved users will only participate in failed transactions occasionally (for example if they interact with a rogue device or if the devices become out of range during the transaction) and therefore will not observe a significant impact on their good reputation. In contrast, repeatedly misbehaving users will have their reputation consistently degraded, possibly to a point where it becomes questionable to other participants.

Table 5.2 summarises the gains and losses of both participants when M3–M6 are attempted. The table analyses the financial impact from a cost/benefit perspective, thus considering it neutral when the results are delivered and the sharer receives the corresponding payment. The requirement that TwinCoins are used at most once considerably reduces the benefits of misbehaving users. Marginal gains of consumer performing M3 or M6 are the degradation of the sharer reputation. Likewise, both M4 and M5 imply the degradation of the sharer's reputation. M5 also implies financial loss for the sharer given that to be effective, the sharer must deliver only the first coin and therefore will not be able to claim the credit of the TwinCoin. The gains for the misbehaving sharer are exclusively related with the losses of the consumer, with a financial loss in M4 and a reputation degradation in M4, M5 and M6.

Given that reputation certificates are signed by the CTE and include the issue date,

an approach to implement misbehaving action M7 is to hide a user's lower reputation information by presenting to tentative peers older reputation certificates. Devices are suggested to consult the local reputation system whenever they suspect that another user reputation is older than expected. The outcome can be either a more up-to-date reputation certificate pushed by the CTE to another device's cache, reports of neighbours' experiences with the suspected device on recent transactions or no additional information. However, because the CTE gossips reputation certificates following a biased distribution that privileges those with the lowest reputation, the absence of information is less likely for misbehaving users. In addition, the local reputation system can rapidly disseminate user's reputation as soon as the first misbehaving attempt is detected by one of the participants.

As shown before, misbehaviour actions M3–M6 allow that a user degrades other' reputation, thus facilitating M8. However, the impact is limited and slow, given that the CTE accepts at most one failed transaction from each pair of pseudonyms. This is a fair restriction, given that users are expected to learn from their experiences. Therefore, they will likely refuse to participate in transactions with users that misbehaved in the past.

Bullying by a coalition of users to degrade another user's reputation or credits is considered in M9. To achieve it, the members of the coalition must repeatedly use actions M3–M6 and therefore, simultaneously degrade their own reputation and (possibly) credit. In addition, because the CTE accepts at most one complaint from each pair of pseudonyms, to become significant, either the coalition should be considerably large or the attack should last for time enough to allow all its members to change their pseudonyms.

In misbehaviour M10, a group of users colludes to artificially increase their reputation. This can be achieved by having the users to perform fake transactions, exchanging TwinCoins and claiming the corresponding reputation increase at the CTE. A thwart to this misbehaviour is implemented by having the CTE to apply a commission on every TwinCoin credited to any user, thus enforcing a cost to the increase of the

user reputation. A negative side effect of this approach is that the penalty applies also to fair transactions. In our model, the currency obtained by the CTE on the commissions is re-injected in the system by distributing it by the accounts of all devices whose reputation is above a predefined threshold.

5.3.1 Privacy

The MC3E requirement R5, which addresses user's privacy is enforced by using certified pseudonyms on all interactions between the participants and by allowing users to frequently change them. Establishing an adequate frequency for the change of pseudonym is outside the scope of this work. For privacy reasons, pseudonyms should be changed as frequently as possible. However, a change in the pseudonyms has some impact on the capability to punish misbehaving nodes, given that the remaining users will not be aware of the change. Therefore, a user with a new pseudonym will reset his (possibly bad) local reputation and it will be impossible to associate the user to the reputation certificates cached on the devices. The impact is limited by the CTE awareness of the pseudonym change. In particular, the CTE enforces the transfer of the user' reputation from one pseudonym to the following. In a stricter model, the CTE could decline to issue pseudonym change requests to devices with a low reputation or impose a minimal time intervals between these changes.

5.3.2 Resource Usage Efficiency

The discussion on the energy efficiency cost is sustained on the premises that most of the impact on energy consumption comes from the usage of the wireless interface, when compared with the power drained by the CPU for using cryptographic algorithms and performing the task requested by the consumers (Hill et al., 2000; Madden et al., 2002).

Assuming that contacts with the CTE (in particular *withdraw* and *deposit* operations) are performed at user convenience, the energy impact is mostly dictated by the

messages exchanged during transactions. To amortise its cost, the system can piggyback most of its information in messages required by the MC3E. This is the case for the transfer of the first coin and of the reputation certificates. Exceptions are the transfer of the second coin and local reputation learning operations, where the former forces to the transmission of a single message. The latter is a 1-hop broadcast, thus implying at most one message from each 1-hop neighbour.

Additionally, to the energy impact, the storage space usage also has to be addressed. The dissemination algorithm that the CTE uses to distribute reputation information to the devices should consider that there is insufficient space available to store the full reputation information database. It is anticipated that this dissemination process can be improved. Two possibilities arise. The first one disseminates with a higher probability the list of the devices with the lowest reputation information. However, if the number of the devices on the C3 increases, this list will probably include a number of unneeded reputation information. The second option is to predict future contacts and disseminate information according to these predictions, something that will be address in Chap. 6.

5.4 Summary

Collaborative wireless environments, such as C3, require mechanisms to cope with the inherent lack of trust among its members. The challenges are amplified by the operational environment, claiming for a moderate use of the network due to the limited computing power and energy available. This chapter described a Hybrid Trust and Trade (HTnT) system combining a reputation system with a virtual currency. HTnT contributes for a reputation model where users benefit of the system proportionally to their contribution. In addition, the service encourages the use of pseudonyms that can be frequently changed, reducing the use of unique identifiers that could compromise the user privacy.

HTnT dismisses the requirement of an internet connection by using two forms of

reputation. The first one is supported by a central trusted entity (CTE), that has no interference during the transactions. The CTE is responsible for the dissemination of a list of reputation certificates and for storing the virtual currency of each device. The second form of trust leverages on recent experiences of the participants and enables the exchange of reputation certificates between neighbouring devices.

6

Human Mobility

The knowledge on human mobility is used on pervasive computing environments to model applications (Orlinski and Filer, 2013) and routing protocols (Hui et al., 2011), to harvest computing resources (Conti et al., 2010) or provide network connectivity (Pelusi et al., 2006) to a group of mobile devices. Groups of devices facilitate for example the creation of distributed data stores (Miranda et al., 2007), message passing in delay tolerant networks (Pelusi et al., 2006) and leverage middleware to efficiently find useful devices for resource sharing (Conti et al., 2010).

In the context of the Mobile Collaborative Cloudless Computing Environment (MC3E), contact patterns obtained from the observation of user mobility are expected to improve the dissemination of reputation information of the Hybrid Trust and Trade System (HTnT). HTnT is component responsible for enabling trust and incentive on Mobile Collaborative Cloudless Computing Environment (MC3E) participants.

Contact patterns are usually estimated from observations of multiple metrics on a population of individuals by applying a statistical fitting on the data. The goal is to find a statistical distribution that provides a good approximation to the metrics of interest and that can be evaluated in run-time. One of the most frequently cited metrics is the inter-contact time (ICT), which represents the time interval between two consecutive contacts of the same two peers. ICT is used in multiple applications and modelled by several mobility models using a power law distribution (Karagiannis et al., 2010; Lee et al., 2009; Boldrini and Passarella, 2010).

However, considering only ICTs limits improvement strategies for HTnT and other

classes of applications, since only pairs of devices are considered. A new metric is needed to effectively model the neighbourhood size and the recurrence of contacts of group members, given its ephemeral nature, the metric is named temporal community. Such a metric would pave the way to improve the dissemination of reputation information between peers that require cooperation. By improving the distribution of the global reputation information dataset while minimizing data redundancy and increasing data availability.

Unfortunately, the design of such a metric is severely constrained by the large amounts of mobility data required to give statistical relevance to any modelling effort. To achieve this we start by analysing a dataset of the eduroam wireless network site on the Polytechnic Institute of Lisbon. The dataset contains all the records produced between 2005 and 2013 by the 76479 devices that accessed at least one of the network's 239 access points (APs).

The dataset is used to determine the dimension and the patterns of repetition of meetings between groups of devices of any size and presents statistical distributions that can be used to model the group contact probabilities. The chapter shows that the extracted statistical distribution fits a *Pareto* distribution for most of the data, with different parameters according to neighbour size and recurrence period.

Leveraged by these results, the chapter describes a ranking algorithm that can be used to predict future contacts between pairs of devices. We applied the algorithm to a mobility scenario extracted from the same data but with a different methodology and year, and also to a trace of GPS positions of Taxis in Rome, and found that all the different environment share the same statistical properties, allowing the ranking algorithm to improve the odds of knowing which devices will be in range in a future day. This knowledge will be used to bias the dissemination of the global reputation information dataset that is sent to each device during contacts with the Central Trusted Entity (CTE).

The chapter is organised as follows. The mobility of users and devices is first introduced in Sec. 6.1. Section 6.2 presents the dataset used together with an analysis

on the impact in human mobility by the introduction of smartphones and MobIPLity a tool that enables the generation of mobility scenarios using the dataset previously presented. Section 6.3 analyses the most common metrics used to characterize mobility from the generated scenarios and compares its results to the ones obtained from other mobility models. Observed temporal communities are analysed in Sec. 6.4 together with our efforts in modelling contacts recurrence into statistical distributions.

6.1 Mobility Extraction

In most mobile applications, HTnT included, the device mobility is dictated by the movement of their owners, what makes of human mobility a key factor influencing the evaluation process of these applications. This evaluation resorts in many cases to network simulators, putting to test an implementation of the application against abstractions of the environment, traffic and device movement. Therefore, the reliability of the experiments performed by network simulators is strongly influenced by the capability to reproduce reality of each of these abstractions. Despite the limitations, simulations play a fundamental role in the performance evaluation of mobile applications and protocols as they permit circumventing the difficulties in deploying large scale and long term real experiments. Most simulators use for movement simulation a model that tries do portrait human mobility by reproducing a number of metrics.

6.1.1 Characterisation of Human Mobility

The mobility of individuals has been characterised along spatial, temporal and social axis (Karamshuk et al., 2011). Spatial axis considers aspects like node density and distance, portrayed by metrics like *jump size* (sometimes referred as *flight*) and *inhomogeneity*. Jump size characterises the average distance travelled by users and is affected by the characteristics of the area, for example, by the distance between buildings. Trace-based mobility models have been modelling jump sizes with either log-

normal (Kim et al., 2006) or truncated power law distributions (Boldrini and Passarella, 2010; Lee et al., 2009). The *inhomogeneity metric* aims at evaluating the dispersion of the individuals on the physical space, in order to highlight hot-spots, which the proposers (Schilcher et al., 2008) consider to be a natural characteristic of human mobility. The variation of the Random Way-Point presented in (Gyarmati et al., 2008) and the Disaster Area (Aschenbruck et al., 2007) are good examples of mobility models enforcing a heterogeneous node distribution. A lower inhomogeneity value is expected from random distributions, while a higher value shows that users are creating groups, formed by nodes placed in popular locations.

Time-varying properties of human mobility characterize patterns such as workday/weekend variations and *pause times*, i.e. the time spent on a specific place. Spatial properties are usually tied with temporal ones, associating the time to the distance travelled between two points, defining metrics such as *speed*.

The social axis characterises the meetings between individuals. In combination with the temporal axis, they contribute to determine how long or how frequently two or more individuals meet. Multiple models with a strong focus on the social relationships established between individuals have been proposed. Metrics considered include attraction (found for example in (Boldrini and Passarella, 2010)) but also repulsion. Both properties are explored in (Fischer et al., 2010) by combining the modelling of relationships with individual walks and group trips. The *inter-contact time* (ICT), defined by the time interval between two consecutive contacts of two individuals, is a frequently used metric to relate the temporal and social axis. Trace-based synthetic mobility models frequently model ICT using a truncated power law distribution (Lee et al., 2009; Karagiannis et al., 2010; Chaintreau et al., 2007).

6.1.2 Modelling of Human Mobility

The modelling of human mobility has been pursuing two approaches: pure synthetic and trace-based synthetic mobility models. In pure synthetic mobility models

nodes move according to some predefined statistical function. The commitment of the rules defined for synthetic models in the replication of observed user movement patterns vary but this model has been criticised for its inability to reproduce human movement patterns, when evaluated by metrics like inhomogeneity (Bettstetter et al., 2003). As an example, consider random waypoint (Johnson and Maltz, 1996), one of the most popular synthetic mobility models, where nodes unrealistically cycle between moving in straight lines to a random location and pause, both for random amounts of time.

Trace-based synthetic mobility models derive statistical distributions from observations of user movement, thus trying to mirror properties observed in real traces of human mobility (e.g. (Gyarmati et al., 2008; Lee et al., 2009; Boldrini and Passarella, 2010)). Traces are provided either by volunteers, which make their location available, or by a third party performing passive observation. Unfortunately, the number of trace-based samples made publicly available is scarce, present a small time span and/or number of users.

To circumvent the limitations of the pure and trace based synthetic mobility model classes, this work proposes a new class, named *trace-set mobility models*. The novelty of this class is that it creates mobility scenarios exclusively from data observed in real traces, disregarding any statistical approximation. As a contribution for the creation of this class, we presents a dataset composed of the observation of the 76479 wireless devices that connected to the eduroam network of the Polytechnic Institute of Lisbon, Portugal (IPL) between 2005 and 2013.

6.2 WiFi Dataset

In order to extract the multiple metrics of human mobility we obtained a large data-set of WiFi access records. In this dataset it is possible to observe the changes on mobility patterns of users throughout the years. The changes of mobility patterns can be tied to the type of the device or with inherent user behaviour changes.

Different classes of mobile devices have contrasting usage patterns. For example, comparing to laptops, a smartphone tends to present a usage pattern with smaller but more frequent accesses, possibly while its user is moving. On the other hand, the reduced dimensions of these devices limit not only its storage capabilities but also the user interface, in particular data input. The differences in the usage pattern of the devices are expected to be reflected on the usage pattern of wireless networks.

Knowledge about the use of Wi-Fi networks is an important tool for multiple players in the research and development arena. Developers benefit from the knowledge of the communication patterns and the attention time to create applications more proactive and to make network use more efficient. Researchers can use this information to develop new protocols and to improve the current synthetic mobility models. Finally, system administrators will see their planning and management tasks simplified.

Eduroam is a Wi-Fi network connecting educational institutions of 71 territories.[1] One of the most important feature of this network is the transparency that it offers to its users, since access is conditioned only by the user's home institution. The academic environment exhibits a natural propensity to the use of a large number and a wide range of distinct devices, making it an appealing environment for the study of wireless networks. This work uses the access point (AP) access-logs from the eduroam branch deployed in the different faculties and services of the Polytechnic Institute of Lisbon, Portugal (IPL) between 2005 and 2013.

The analysis of this dataset follows in two directions. In first place, a set of information on the evolution of network utilization is presented, measured by the number and type of the devices. Secondly, we will show that the emergence of smaller devices resulted in a non-negligible change in the use pattern of wireless networks. Overall, these contributions confirm with evidence a number of unsupported assumptions, concerning the increasing use of wireless devices and user mobility, that can be found in the literature and which may influence not only Hybrid Trust and Trade (HTnT) but also the development and management of other network applications/environments.

[1]See "eduroam homepage" at: https://www.eduroam.org/index.php?p=where (Last Accessed: 19 April 2015)

Figure 6.1: Location of IPL sites

6.2.1 Environment Characterization

The dataset used in this study is composed by the access records of all Access Points (APs) of the eduroam Wi-Fi network of the Lisbon Polytechnic Institute (IPL) generated between January 1, 2005 and December 31, 2013. A total of 76479 devices and 45363 distinct users accessed the network during this time frame, producing about 43 million records, an average of 9.2 access requests per minute.

IPL is the 7th largest teaching institution in Portugal with approximately 1300 teachers and 15000 students registered on one of the 88 bachelors and masters offered. IPL is distributed over 10 distinct sites in Lisbon metropolitan area (see Fig. 6.1). The eduroam network is supported by 236 Cisco Systems APs, covering a total of 26 buildings and inter-building areas. Records are originated from all the users accessing the network, thus also including visitors from other institutions.

Figure 6.2 depicts the evolution of the number of APs, distinct users and devices. The growth of the AP number is justified, in the vast majority of the cases, by the need to increase the network capacity in order to satisfy the demand. The figure also shows

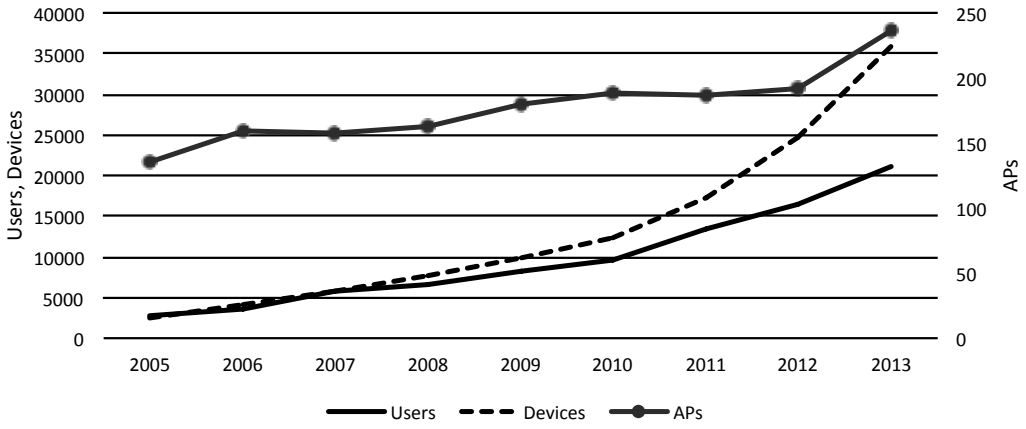


Figure 6.2: Devices, users and access points

a continuous growth of the number of users and devices although at distinct rates, specially since 2010. This is coincidental with an increase in the sales of smartphones observed at the national level and confirms our expectations that the number of users accessing the network with more than one device has been increasing.

The proportion of devices per manufacturer of the wireless network interface is depicted in Fig. 6.3. The manufacturer of each wireless network interface was obtained from the *Organisationally Unique Identifier* (OUI) component of its link layer address. Results are approximate, as link layer addresses can be forged and the OUI may not be respected by some manufacturers. Figure 6.4 depicts the total dispersion of the devices by the most popular operating systems. These results are only estimates given that: $i)$ users are free to change the data sent by their DHCP client and, $ii)$ recent Apple devices don't send *vendor* information. Devices with multiple operating system capabilities are represented once per operating system detected.

The DHCP (Droms, 1997) message fields *vendor, parameter request list* and *hostname* were used to identify the device operating system. Unfortunately, no DHCP records could be found before 2009. Results for years between 2005 and 2008 include only the devices that connected at least once since 2009. This strategy allowed the identification of 65160 (85%) of the 76479 devices that connected to the network between 2005 and 2013.

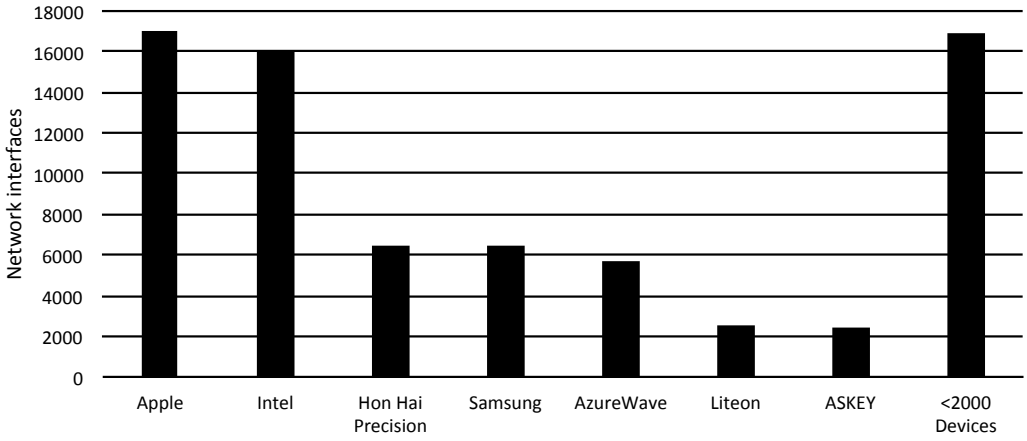


Figure 6.3: Wireless network interface manufacturer

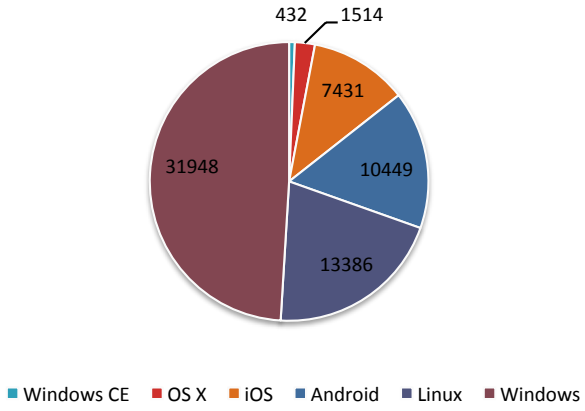


Figure 6.4: Detected operating systems (by ascending order)

Our study arranges devices in two classes: *Small Mobile Devices* (SMD) include those that tend to be always on, are small and can be used on the move. Examples of SMDs are smartphones, PDAs and tablets, using Windows CE, iOS and Android. The second class, *Laptops*, group the larger devices usually running over a classical operating system (Linux, Windows or OS X). The class of each device was determined by its operating system. Figure 6.5 shows that the share of SMDs is increasing rapidly, reaching more than 20% in the last year of our study.

Data analysis is centred on RADIUS (Rigney, 2000) logs. This service is responsible for the authentication of all eduroam network access requests. The RADIUS service generates a log entry every time a user associates or de-associates to an AP, as well as

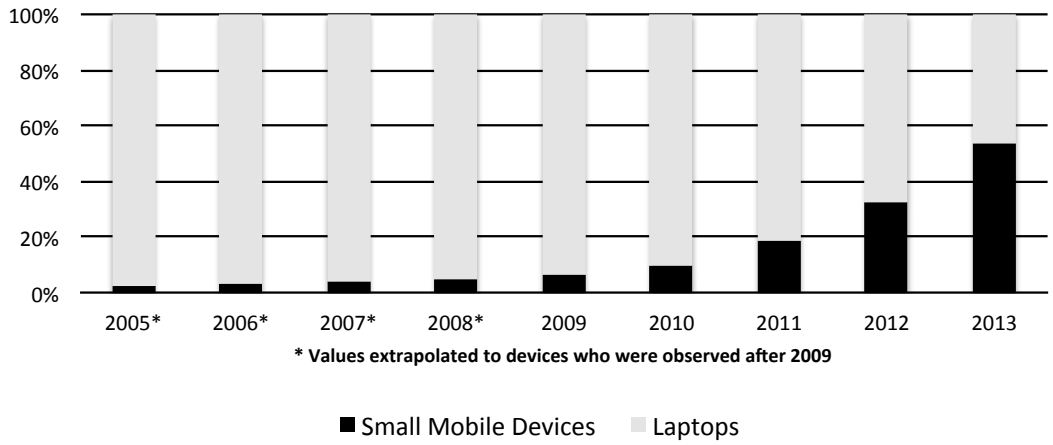


Figure 6.5: Laptops vs small mobile devices

when keep-alive messages are exchanged. Keep-alive messages confirm the presence of a user already associated. Log entries reproduce the RADIUS session concept thus considering the association of each user to a single AP and therefore ignoring user mobility. We will use the term *session* to refer to these records. Log entries contain the device MAC address, AP, user name, session start and stop times and total traffic sent and received during the session.

In ideal conditions, each session should represent the association of a device to an access point. However, the number of sessions observed is slightly amplified due to: $i)$ automatic handover between APs, triggered by variations in signal strength; $ii)$ incompatibilities between client drivers and protocol versions running in the AP and; $iii)$ operating system energy saving mechanisms that may turn off the radio interface when it is not in use. Interpretations of the results which rely on the number of sessions should therefore be made with some caution and take into account these factors. To mitigate some obvious anomalies, logs have been edited by:

- merging in a single record consecutive sessions between the same device and AP with an interval of less than 5 seconds. These sessions are attributed to network card or driver problems;

- removing concurrent sessions of the same device to distinct APs. This is an im-

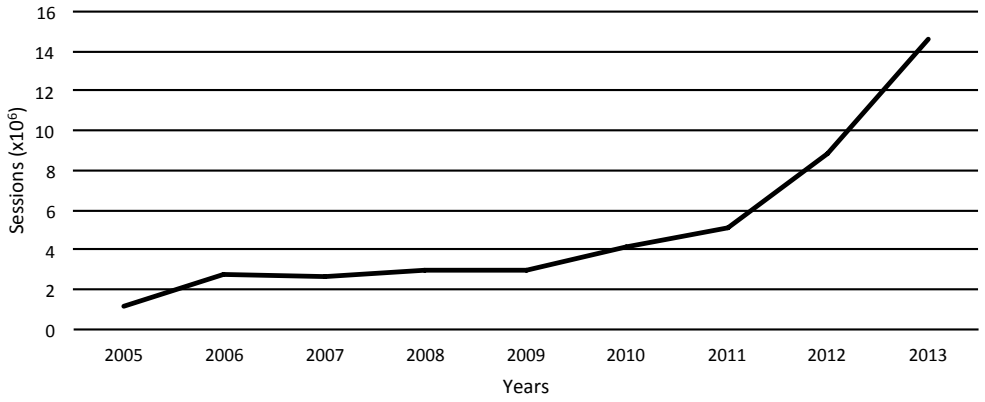


Figure 6.6: Sessions

possibility that can only be explained if the device did not disassociate correctly from one AP before associating to the next and the former artificially defined the session stop time upon a timeout. In this case, the session stop time of the earliest session was corrected to happen immediately before the start time of the latest;

- removing sessions with the stop time equal to the start time. Sessions with these characteristics are created when a user has some problem while connecting to the network, although the network considers the user authenticated (thus creating the RADIUS record).

The evolution of the total number of sessions with time is presented in Fig. 6.6. However, the temporal evolution on the absolute number of sessions must consider the gradual capacity growth of the eduroam network (cf. Fig. 6.2), which in case of user mobility can increase the number of sessions established on the same path but taken in different years.

Figure 6.7 depicts the number of distinct devices that appear on RADIUS logs per day. As expected, the plot exhibits an irregular pattern consistent with the different activity levels that can be found on workdays, weekends and summer and winter breaks in the campus.

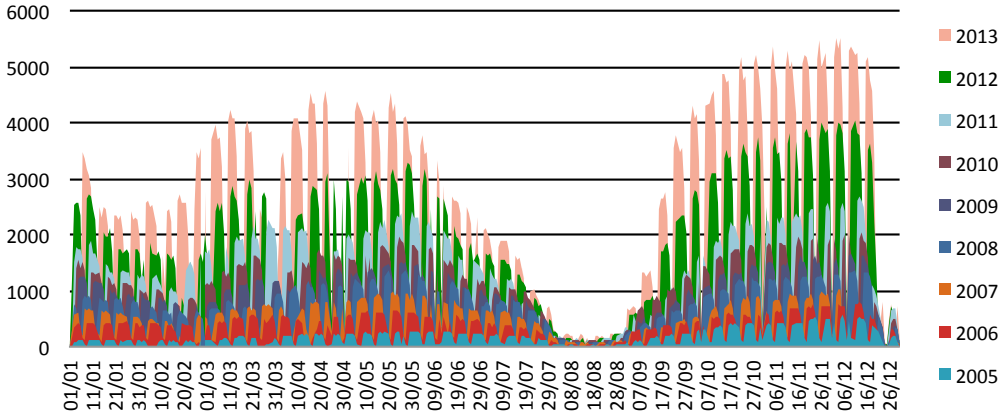


Figure 6.7: Wifi Devices Connected Per Day

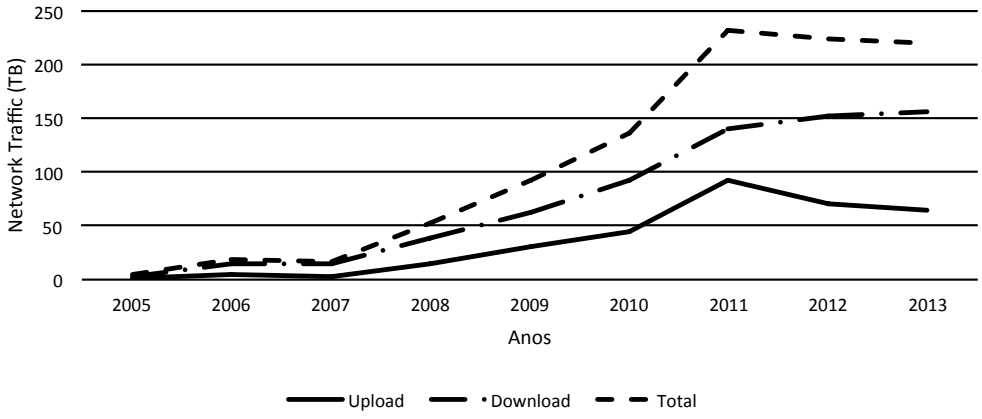


Figure 6.8: Network traffic

6.2.2 Dataset Generic Analysis

Figure 6.8 shows the evolution of the network traffic produced by the users. A steady increase in all types of traffic can be observed until 2011. This is consistent with the increasing number of users and devices registered. However, in 2012 there is a decrease in the traffic produced by users, that results from a decrease of the growth rate of downloads and from a reduction in the absolute value of upload traffic. This results is in clear contrast with the significant growth in devices and users in 2012 depicted in Fig. 6.2.

The introduction of flat rates for data traffic by 3G and 4G cellular operators, a hypothetical overload of the network infra-structure or the introduction of new network

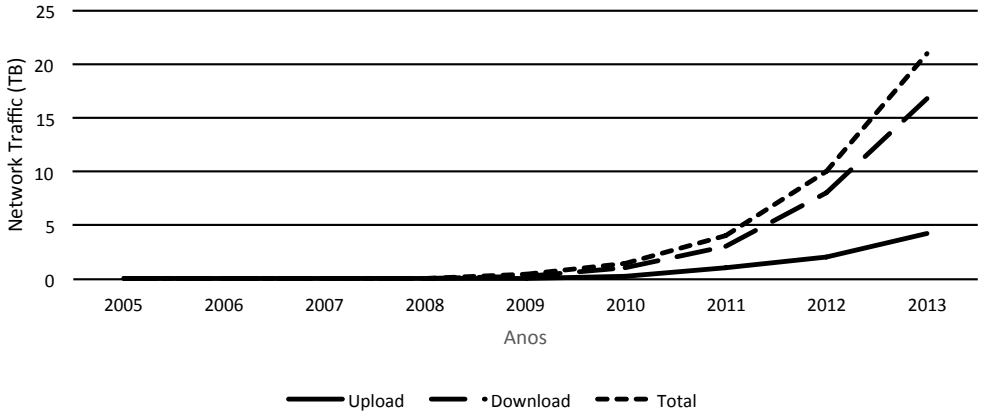


Figure 6.9: Network traffic of small mobile devices

policies could explain this change. However, the former doesn't justify the difference in the direction assumed by the curves of upload and download and no other evidence of the latter was observed. Instead, the decrease of the traffic in 2012 and 2013 is attributed to a change in the usage pattern of wireless devices which lost their role of information producers. This conclusion is supported by the increasing proportion of small SMDs, which do not favour the entry or sharing of data (for example through peer-to-peer networks) and by Fig. 6.9, which shows the evolution of the network traffic produced exclusively by SMDs. The figure confirms that, in contrast with the overall results, SMDs network traffic continues to increase at a considerable rate although its weight (about 10% for 2013) is still insufficient to obfuscate laptops decaying network usage.

A relevant aspect is that, as depicted in Figs. 6.6 and 6.10, despite the traffic reduction, the total number of sessions and the accumulated usage time continues to increase. It is therefore safe to conclude that the time during which the devices are associated with an AP but do not use the network is increasing. Additionally, there is a steady decrease in the average session duration per user since 2008, which suggests that the increase in the accumulated time of use of the network is fully supported by the growing number of users.

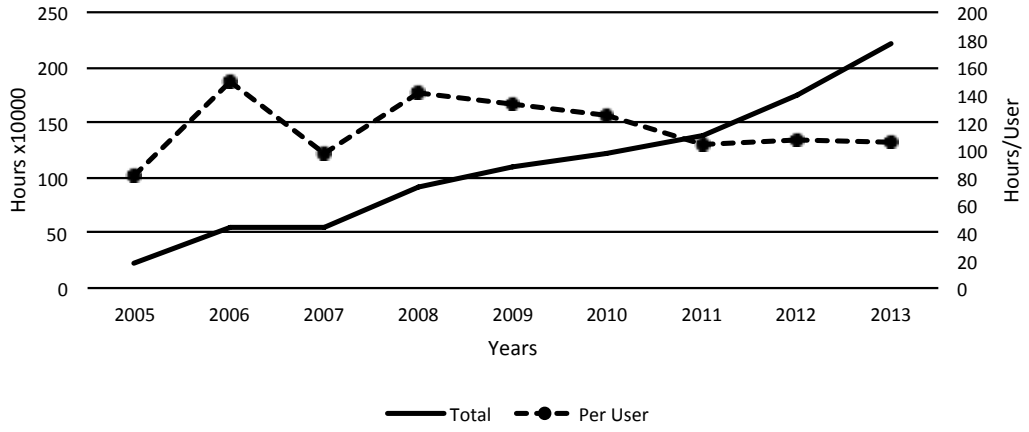


Figure 6.10: Session duration

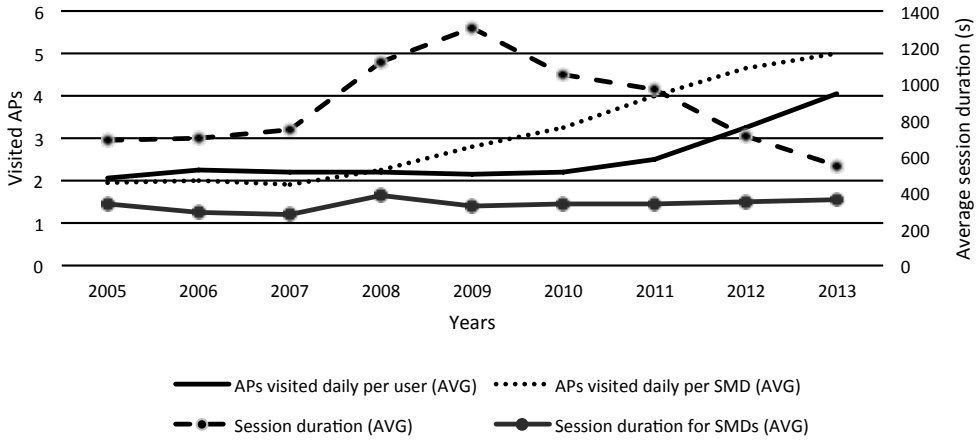


Figure 6.11: Yearly evolution on the number of visited APs and session duration

6.2.3 Evolution of User Mobility

To evaluate user mobility, this study focus on the number of distinct APs visited daily by each user. Fig. 6.11 suggests that user mobility can be partitioned in 3 distinct periods.

6.2.3.1 2005 to 2007

The period between 2005 and 2007 is characterized by the stability of both the average number of access points visited, and the session length. That is, the devices tend to

be fixed, with no variations in the use pattern being observed during this period. This stability is consistent with the ratio of 1 to 1 between users and devices (cf. Fig. 6.2), the negligible weight of SMDs (Fig. 6.5) and the slow traffic growth rate suggested by Fig. 6.8.

It is also noticeable that the group of SMDs has a lower average number of visited APs, this is barely meaningful given that SMDs aren't even 5% of the total number of devices. However the stability of this value suggests that the class of SMDs wasn't as mobile as we could think of it, by this time most SMDs were PDAs (Personal Digital Assistants). This could suggest that the wireless technology at that time lacked the power efficiency of today wireless interfaces available on SMDs, having a large impact on battery, forcing users to turn on the wireless interface only under controlled use, turning it immediately off after using the application that required network access.

6.2.3.2 2007 to 2009

The second period (between 2007 and 2009) is characterised by an increase of nearly 100% of the average duration of sessions but without no change in the number of visited access points per user. Fig. 6.8 shows that this period coincides with the increase of traffic, although the number of users continues to grow at an almost linear pace. The short distance between the number of devices and the number of users (cf. Fig. 6.2), and the stability of the number of visited APs, suggests that this period is uniquely characterized by an increase in the volume of IPL eduroam network use.

6.2.3.3 2010 Onwards

The year of 2010 marks the beginning of a new use pattern where users connect to the network at a larger number of locations, although by shorter amounts of time. In this period, the average session duration falls progressively to values that in 2013 are lower than the ones from 2005. Simultaneously, the average number of visited APs increases by more than 50%. This result confirms our expectation that a significant

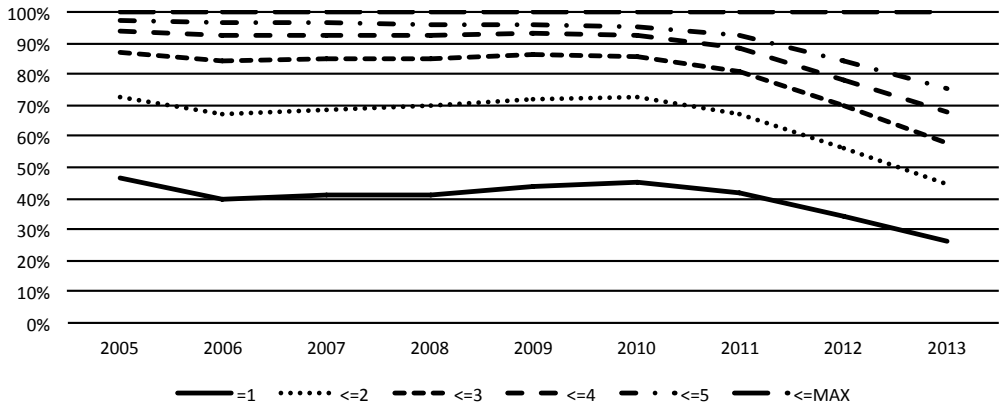


Figure 6.12: Distinct APs visited per user daily

Table 6.1: Maximum number of distinct APs visited by a user in a single day

2005	2006	2007	2008	2009	2010	2011	2012	2013
18	22	25	25	27	28	32	52	48

change is taking place in the wireless network usage pattern. This change is attributed to the wider deployment of SMDs, what is supported by noting that this is also the period in which the ratio between devices and users increases, and in which the ratio of SMDs gains relevance. It should also be noted that SMDs exhibit a pattern that differentiates from the average results, both when considering the number of visited APs (considerably more than the average) and the session duration (considerably shorter).

Figure 6.12 deepens the study on the mobility of users by analysing the distribution of the number of access points visited daily per user. The results are in line with previous findings, with the values being stable until 2010. During this period, less that 10% of the users visited 4 or more APs in a single day. However, from 2010 onwards, it is possible to observe a steady growth in the number of users with more than 5 APs visited daily. In 2013, these were already more than 20% of the users. These results confirm the suspicions of an increasing user mobility. Although not statistically significant, and mostly as curiosity, Tbl. 6.1 shows the evolution of the maximum number of distinct APs visited by a single user in a day.

On the other hand, the decrease in traffic and on the average session duration (resp.

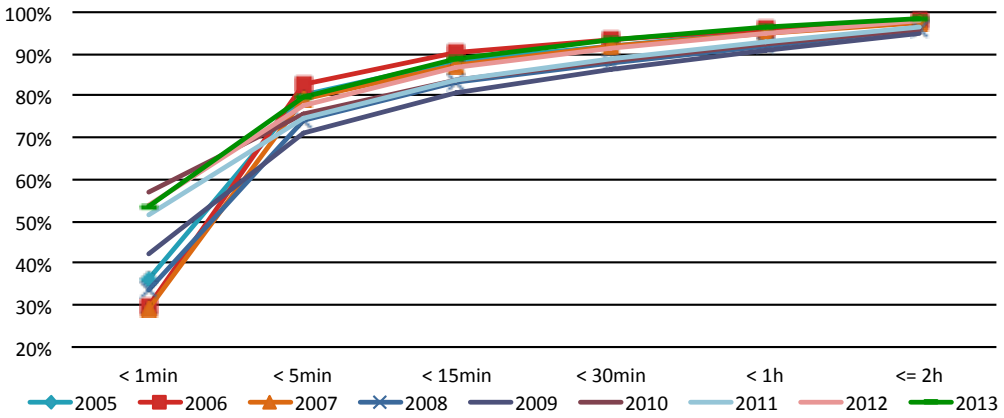


Figure 6.13: Detailed session duration

Figs. 6.8 and 6.11) raises questions about the usefulness of these sessions. Figure 6.13, which depicts the distribution of mean session duration, exhibits two distinct patterns. Between 2005 and 2009, approximately 80% of the sessions had a duration of less than $5min$, with more than 50% in the range between $1min$ and $5min$. In contrast, from 2010 onwards, more than 50% of the sessions are shorter than $1min$. Interestingly, the distribution of longer sessions is kept relatively stable across the study period.

The non negligible number of very short sessions ($< 1min$) can be interpreted in several ways. On one hand, it could result from protocol negotiation problems that force the devices to frequently restart the sessions. However, it would be expected that the proportion of sessions with problems remained constant over time, or even decreased as a result of technological improvements and error elimination in protocol implementations. Additionally, the attribution of these sessions exclusively to protocol issues doesn't justify the reduction in traffic, the increase on the number of daily visited access points by users or the decrease on the utilization time per user (Cf. resp. Figs. 6.8, 6.12 and 6.10).

We claim that, instead, these very short sessions can be attributed to users that keep their SMDs with the radio interface enabled while moving between different campus locations, establishing connections with access points while on the move. This conclusion is supported by relating the two patterns (2005-2009 and 2010-2013) observed in

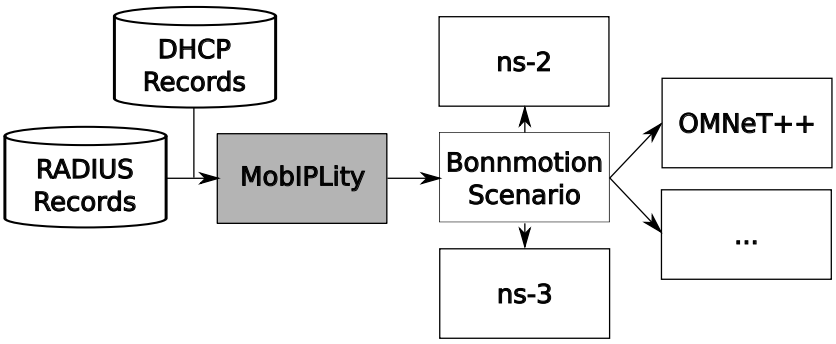


Figure 6.14: MobIPLity Work-flow

Fig. 6.13 with the introduction of SMDs, that since 2009 is gaining visibility. In addition, Fig. 6.11 shows that the growth in the number of visited APs is not reflected in the session duration of SMDs. Interestingly, this could also hint changes in laptop use behaviour. Apparently, users are not replacing them by SMDs, but at the same time use laptops in new ways. These findings suggest a more judicious separation of activities between SMDs and laptops, with the former being used while on the move (thus the larger number of APs visited) and the later for other kinds of tasks for which SMDs are not suited.

In summary, these results confirm an increase in the number of users and devices, with the latter growing at a faster pace. Results also show that mobility has increased with users connecting to an increasing number of access points daily. The combination of both results suggests that there is an ongoing change in the use pattern of wireless networks. Users now tend to access to wireless networks through two devices, with one turned on even when the user is moving. However, the increase of the number of devices and connectivity is not reflected in traffic, which tends to decrease (in spite of an increase in mobile devices).

6.2.4 MobIPLity: Mobility Scenario Generator

To create the prediction algorithm we need to analyse the size and duration of temporal communities. MobIPLity, is a mobility scenario generator developed in the scope of this work that uses the IPL eduroam dataset presented above to create realistic

mobility scenarios.

As depicted on Fig. 6.14, MobIPLity combines records produced by the DHCP and RADIUS services to create mobility scenarios that closely reflect observed user behaviour. In MobIPLity, DHCP records contribute with the identification of the device type and RADIUS identifies the participants and the moment of each association/dissociation event. The geographical coordinates of the access points contribute with the estimate of the location of each participant at each association/disassociation moment. Scenarios are produced in bonnmotion format (Aschenbruck et al., 2010), a popular mobility scenario generator capable of interacting with numerous network simulators.

The following sections present the algorithm being used for extracting the traces from the information above.

6.2.4.1 Trace Generation

The MobIPLity trace-set mobility model is created from a set $E \subseteq D \times A \times \{IN, OUT\} \times T$ where D is the set of wireless devices, A the set of access points of the network annotated with their geographical coordinates and T are time stamps. The set is populated with 2 events $(d, a, IN, t_1), (d, a, OUT, t_2)$, for each RADIUS log record made available by the eduroam network of IPL. In these events t_1 and t_2 are time-stamps reflecting respectively d's association/disassociation to AP a.

Let $E_d \subseteq E$ be the subset of E containing all the events recorded for device d. The set E_d is expected to respect two invariants: $i)$ devices are always associated with an access point before being disassociated from it, i.e., $\forall (d, a, OUT, t') \in E_d, \exists (d, a, IN, t) \in E_d : t \leq t'$; and $ii)$ in any point in time, a device is associated at most to one access point, i.e., $\forall (d, a, IN, t), (d, a', IN, t') \in E_d \wedge t \leq t', \exists (d, a, OUT, t'') \in E_d \wedge t \leq t'' \leq t'$. It should be noted that invariant $i)$ is trivially assured by the access points software and invariant $ii)$ by the corrections applied to the RADIUS logs that have been outlined in Sec. 6.2.1.

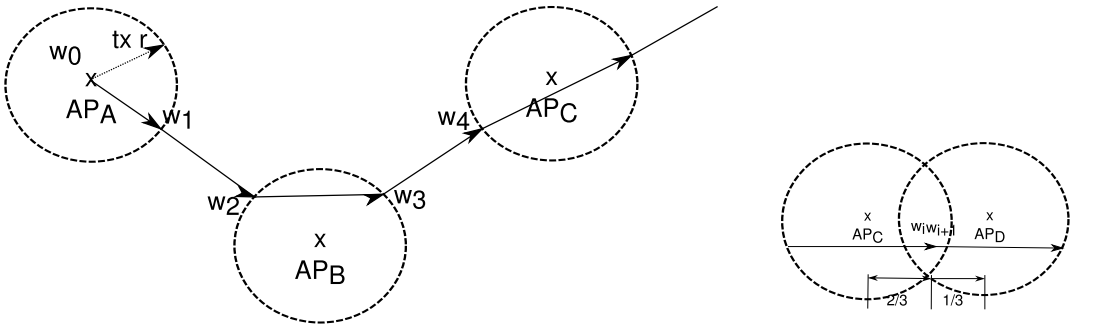
We define $E'_d = e_{d,0}, e_{d,1}, \ldots, e_{d,n}, d \in D, e_{d,i} \in E_d, i > 0$ as the temporally ordered set of events for device d. It should be noted that according to invariants $i)$ and $ii)$, $e_{d,2j}, j \geq 0$ are events of type *IN* and, conversely, $e_{d,2j+1}, j \geq 0$ are all events of type *OUT*.

A trace $W_d = w_0, w_1, \ldots, w_{2n-1}, n \geq 1$ for some device d, is defined as a sequence of waypoints $w_i = F(e_{d,2j+i}), 0 \leq i \leq 2n-1, j \geq 0, e_{d,2j+i} \in E'_d$. The waypoints are defined by a geographical coordinate and a time stamp, returned by a function F applied to consecutive events (not necessarily starting in $e_{d,0}$) in E'_d. The output of function F depends of:

- the position of the waypoint on the trace;
- the type (IN, OUT) of the event;
- the transmission radius estimated for the access point;
- the coordinates of the access point;

The general case is depicted in Fig. 6.15a. w_0 is set with the time stamp of $e_{d,2j}$ and the coordinates of the access point in this event. Subsequent transformations of pairs of events on pairs of waypoints $w_{2i+1}, w_{2i+2}, i \geq 0$ will return coordinates overlapping a vector $\overrightarrow{AP_A AP_B}$, with AP_A, AP_B being the coordinates of the access points in the corresponding events $e_{d,2j+2i+1}, e_{d,2j+2i+2}$. The final locations of the waypoints is dictated by the transmission radius of the access points, as w_{2i+1} (resp. w_{2i+2}) will be placed at the intersection of the vector with the transmission radius of AP_A (resp. AP_B). Time stamps of w_1 and w_2 are copied from the corresponding events. Notice that, according to the definition of E'_d above, events $e_{d,2j+2i+1}, e_{d,2j+2i+2}$ are respectively an *OUT* and an *IN* record, thus signalling the moment at which d abandoned the area covered by AP_A and the moment at which d associated with AP_B. The algorithm is successively repeated for each pair of events and waypoints.

The handling of the particular case occurring when the coverage area of two consecutive access points visited by the device intersects is depicted in Fig. 6.15b. The



(a) With disconnections (b) Overlapping coverage

Figure 6.15: Trace extraction examples

algorithm reflects the conservative approach of wireless interface drivers. The two waypoints receive the time stamp of the *IN* record and are set at 2/3 of the distance between the access points. This model reflects the expected conservative behaviour of the driver of performing one hand-off only when its benefits become evident.

6.2.4.1.1 Trace Termination Conditions The conditions for terminating a trace are motivated by the need to signal the cases where the device abandoned the network, with the users moving to locations MobIPLity is unable to track and which would be otherwise represented by very slow movements that were not effectively observed. Traces are terminated at an *OUT* event by creating a waypoint with the coordinates of the access point. MobIPLity identifies two conditions for interrupting a trace:

- when the speed for traversing the distance between two consecutive access points falls below some threshold;
- when two consecutive connections to the same AP exceed a time threshold;

Both conditions are triggered by thresholds that by default are set to $0.5ms^{-1}$ and $120s$ respectively, but can be configured according to the user preferences. Expectations are that these thresholds are sufficient for identifying the cases where the user abandoned the campus (for example to go home) in one location and re-entered it at a different (first condition) or at the same gate (second condition). In these cases, a new trace will be started from the next IN event for the same device.

Table 6.2: Trace extraction options

Parameter	Description
Start/End Date	Not before/after dates of the records
# Devices	Number of devices
Device' Type	Laptops/SMD/Any
Points	Minimum number of APs in a trace
Duration	Trace duration
Location	School for extraction
Axes	2D/3D representation
Warm Up	Duration of the warm up period
Cool Off	Duration of the cool off period
Enhanced trace	Include terminated traces
AP Range	Radius of the AP coverage
Speed	Min. speed for consecutive APs
Time	Max. time between connections

6.2.4.1.2 Trace Generation Options Traces are extracted from an E set stored at a local database and running the MobIPLity algorithm described in Sec. 6.2.4.1. The algorithm outputs traces in the format used by the bonnmotion mobility scenario generator and analysis tool (Aschenbruck et al., 2010). All access points are consistently positioned using a random factor. The algorithm normalizes the output, shifting the time stamps and creating an initial waypoint for each trace with the location predicted for each device at scenario starting time.

Table 6.2 lists the parameters that can be configured for the generation of each trace. These parameters have been defined in order to facilitate the generation of a large number of trace instances, possibly, with distinct characteristics. Parameters can be arranged in four categories:

The *number of devices*, *device type*, *points*, *duration* and *location* parameters have a direct impact on the metrics that are more frequently referred in the characterisation of the mobility scenarios. Section 6.3 further addresses this aspect, by focusing on the role

of the *location* and *device type* parameters in the generation of distinct mobility scenarios. The algorithm allows to choose from 12 distinct locations, creating mobility scenarios whose areas range from small to medium campus sizes ($0.001km^2$ to $0.063km^2$) and to a metropolitan area ($40km^2$) by considering the aggregation of the locations depicted in Fig. 6.1.

The *AP Range, Speed* and *Time* parameters influence the MobIPLity algorithm. As described in Sec. 6.2.4.1, *AP Range* has impact on the determination of the device location in traces while *Speed* and *Time* dictate the conditions for individual trace termination.

The *Warm Up, Cool Off, Axis* and *Enhanced trace* parameters address the more technical aspects related with the generation of the scenarios. The *Warm up* and *Cool off* parameters ensure that the devices remain active for the entire duration of the scenario. Therefore, MobIPLity exclusively selects for the scenario devices that have visited at least one access point in the *Warm Up* and *Cool Off* periods that respectively precede and succeed the time interval selected for the scenario. Alternatively, the *Enhanced trace* option enables the inclusion of devices that connect/disconnect from the eduroam network during the scenario. Unfortunately, *Enhanced trace* conflicts with a number of network simulators which do not consider device disconnection in their mobility parameters.

Finally, the *start/end date* parameter allows the creation of multiple instances of scenarios with the same characteristics, knowing that MobIPLity will select the first moment after *start date* where all the remaining conditions can be simultaneously satisfied for creating the scenario.

6.3 Mobility Analysis

This section presents and discusses the characteristics of the mobility patterns found on the 2012 subset of the MobIPLity trace-set. This period was chosen in or-

Table 6.3: Overview of the 2012 trace set

	All		Laptops		SMDs	
	IPL	ISEL	IPL	ISEL	IPL	ISEL
Devices	24141	10080	14947	7066	5403	2056
Traces	2075731	985816	1061686	602620	641394	250209

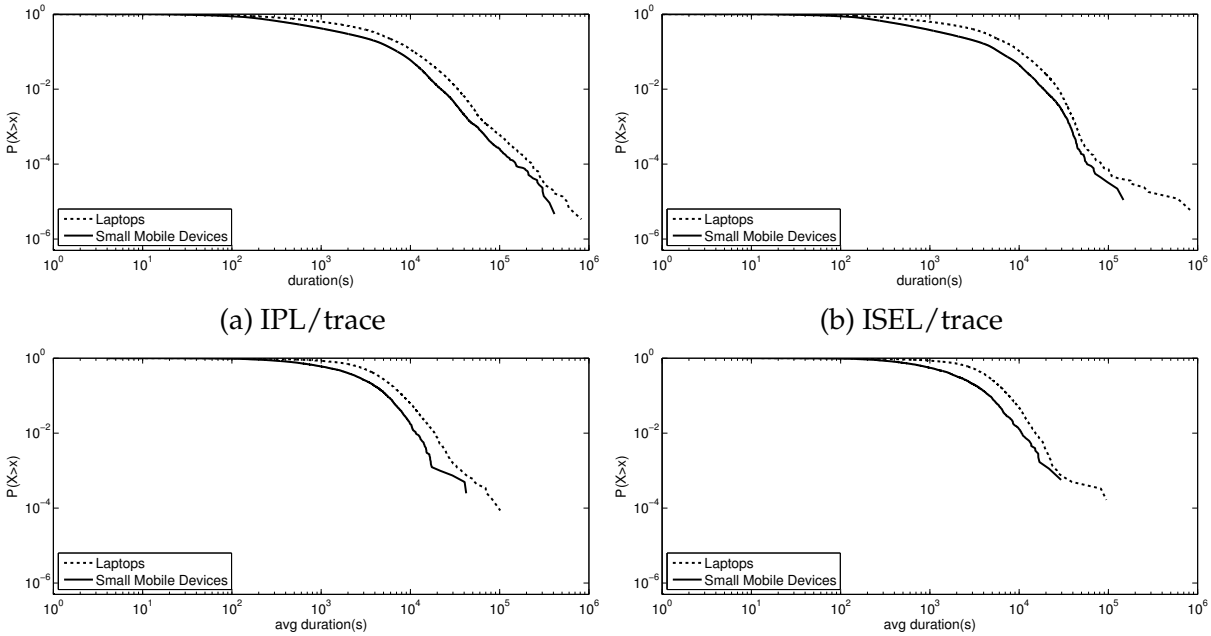
der to allow a comparison with 2013 that will be addressed on the following sections.

Analysis proceeds in side-by-side comparison of the 2 alternative types of devices, allowing to confirm the existence of distinct mobility patterns for users carrying large ("Laptops") and small ("SMD") devices. To further increase diversity, the ISEL and IPL locations are considered. These are the contrasting extremes concerning node density. ISEL is the engineering school of IPL, located in a single site with an area of $0.063km^2$ and provides the largest number of devices from a single location. The IPL location considers records collected from access points at all schools and presents a very small node density as campus are distributed over $40km^2$ of the Lisbon metropolitan area (Cf. Fig. 6.1).

Table 6.3 presents an overview of the dimension of the 2012 and 2013 MobIPLity's trace-set. It should be noted that columns for IPL consider all the institution, thus including ISEL. Still, ISEL accounts with approximately 40% of the devices and of the number of traces.

The first part of the analysis provides an overview of the metrics observed in the traces produced from the complete dataset. The second part refines these results by performing an in-depth evaluation of 2 specific traces.

In the analysis of the complete dataset of 2012, discussion proceeds in two complementary perspectives. The "Per trace" perspective makes no association between the traces. I.e., traces are considered and averaged individually. In contrast, the "Per device" approach first aggregates the traces produced by each device and then proceeds by making an evaluation of the results on a device-by-device basis.



(a) IPL/trace (b) ISEL/trace

(c) IPL/device (d) ISEL/device

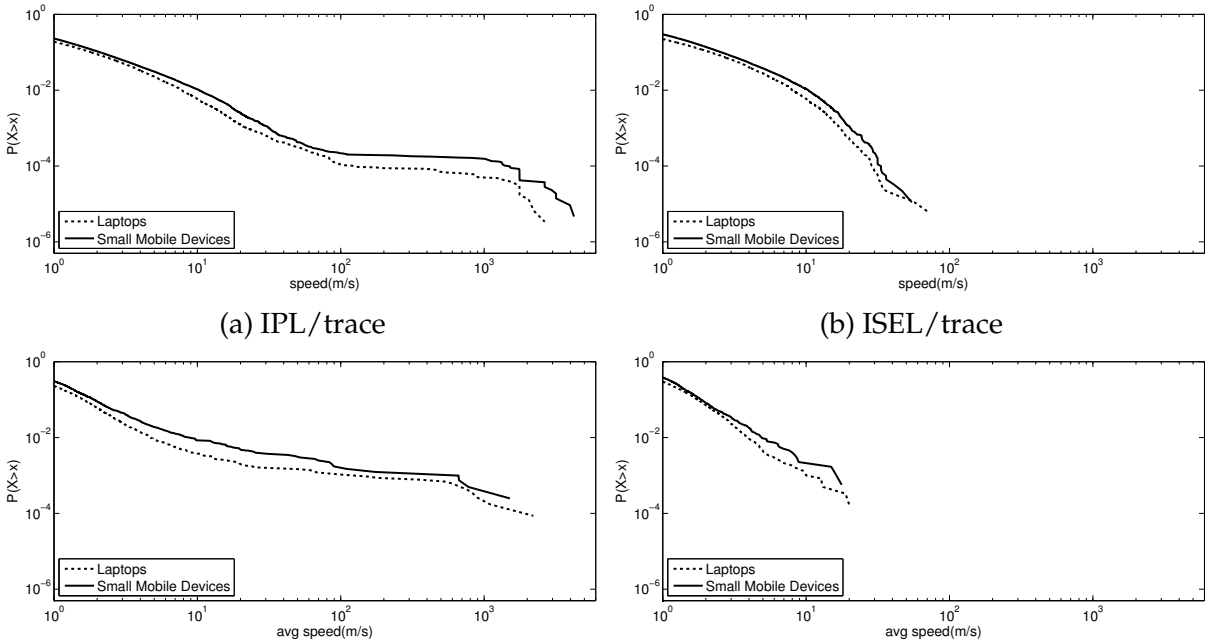
Figure 6.16: Trace duration for 2012 (seconds)

6.3.1 Trace Duration

Figure 6.16 shows the complementary cumulative distribution function (CCDF) of the duration of each trace. Long trace durations were expected, specially for Laptops, due to the coverage provided by IPL eduroam to student dorms.

However, the figure also shows that less than 18% of the traces for Laptops exceed 2 hours and that for SMDs this value further decreases to about 10% of the traces in IPL and 7% in ISEL. These results are confirmed when observing the average session duration of each device on Figs. 6.16c and 6.16d. Such small proportion of "long traces" is surprising. One would expect that the usage pattern reflected the increasing use of mobile devices on the campus and, therefore, that trace durations were consistently higher.

The small duration of the traces, and the consistently lower average duration of SMDs when compared with Laptops, is attributed to the energy-saving mechanisms that can be found on mobile devices. These mechanisms automatically disable the wireless interface when not in use or when the screen is turned off. This is an aspect



(a) IPL/trace (b) ISEL/trace

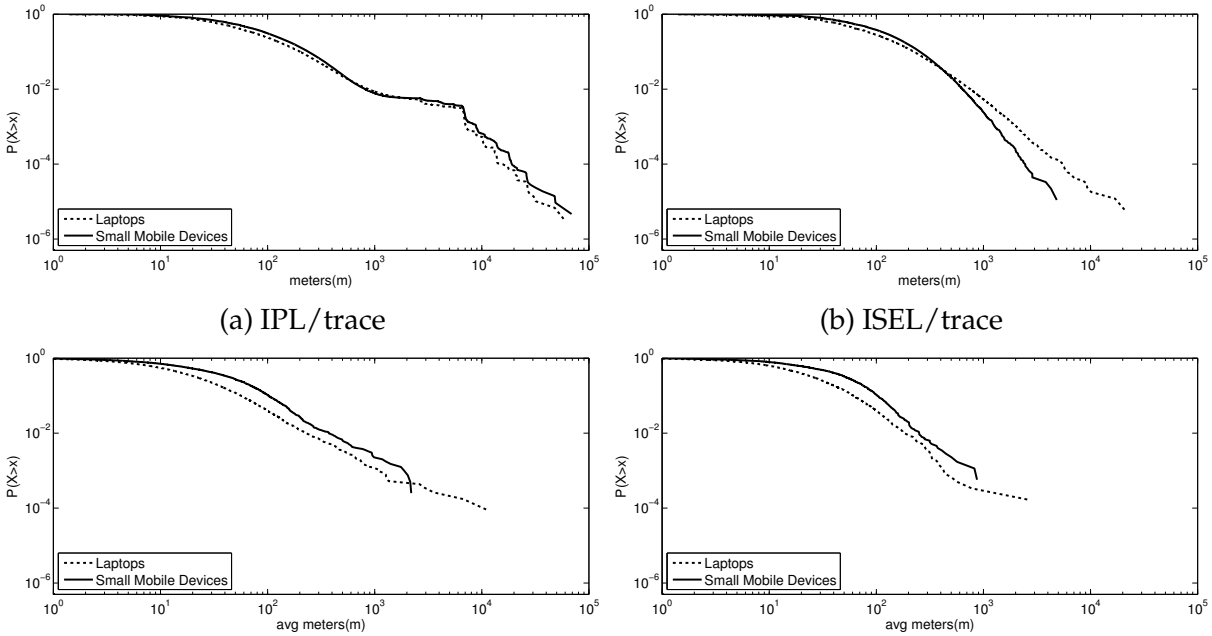
(c) IPL/device (d) ISEL/device

Figure 6.17: Trace speed (ms^{-1})

that has been consistently ignored in trace-based mobility models and is even hard to reproduce in network simulators. However, this feature has a non-negligible impact on the design and evaluation of many protocols and applications for ad hoc and delay-tolerant networks which assume "always on" connectivity of the devices. As a simple example, consider the impact of intermittent connectivity on the route discovery phase (that uses flooding) of many reactive routing protocols for MANETs, such as DSR (Johnson et al., 2001) and AODV (Perkins et al., 2003). A more in depth investigation of the impact of the power saving mechanisms is out of the scope of this work and left as future work.

6.3.2 Speed

Speed plots (Fig. 6.17) exhibit some abnormal patterns of devices moving up to $1000ms^{-1}$. However, these are found on less than 0.01% of the traces and are attributed to the ping-pong effect that results from a combination of the fast roaming of the devices between overlapping APs and the trace generation algorithm used. This is a



(a) IPL/trace (b) ISEL/trace

(c) IPL/device (d) ISEL/device

Figure 6.18: Trace length (meters)

problem that has been observed in other models (e.g. (Kim et al., 2006)) and has a negligible impact as these fast speeds occur for very small amounts of time and distances. It should be noted that a portion of the 15% of the traces with an average speed above the average human walking speed on the IPL trace set can be attributed to users moving between sites, and that 18% of the IPL traces and 30% of the ISEL traces have a speed under $1ms^{-1}$, which simply suggests users walking at low speed.

A comparison by device type shows that SMDs consistently present an average trace speed higher than Laptops. This result confirms the distinct utilisation pattern which can be easily observed in real life, with users operating their SMDs while walking. The "Per Device" perspective for IPL is still affected by the ping-pong effect. However, it is possible to observe a non-negligible number of devices (around 1%) roaming across distinct campus by exhibiting average speeds on the range of $30Kmh^{-1}$.

Table 6.4: Number of samples with no distance travelled

	Location	All	Laptops	SMDs
Per Trace	IPL	2075731/1449669 (70%)	1061686/767041 (72%)	641394/426939 (66%)
	ISEL	985816/682205 (69%)	602620/435860 (72%)	250209/159873 (63%)
Per Device	IPL	24141/6395 (27%)	14947/3486 (23%)	5403/1417 (26%)
	ISEL	10079/1637 (16%)	7066/1092 (16%)	2056/337 (16%)

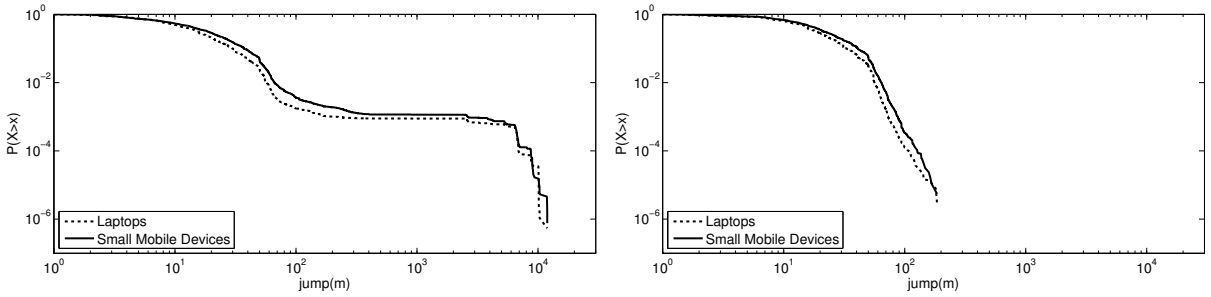
Samples with zero distance/Total number of samples (% of samples with zero distance)

6.3.3 Distances Travelled

The distance travelled is evaluated using two metrics. Trace length is depicted on Fig. 6.18 and measures the length of each trace in meters. The geographical disposition of IPL sites and the roles of some of its members results in some traces obtaining surprising values of 100 Km. However, the "Per Device" averages are more predictable and only reach 11Km for IPL and 800 meters for ISEL.

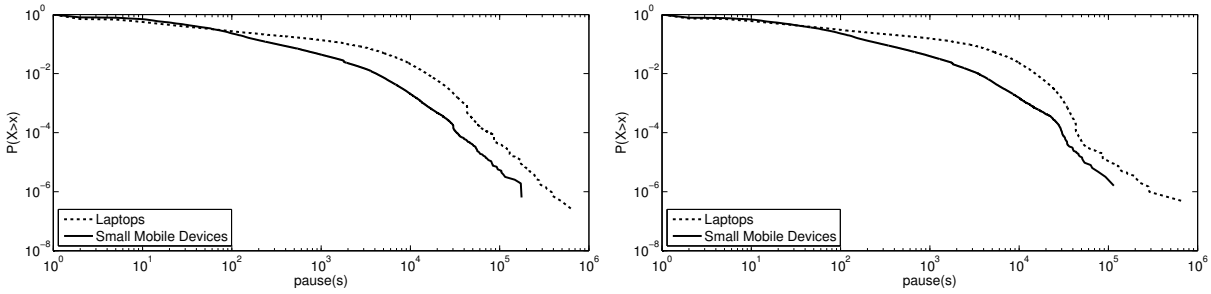
As expected, the higher mobility of SMDs is confirmed by longer average traces in conjunction with shorter durations. However, looking at the complete trace-set of IPL we cannot differentiate between different device type distributions. This is expected, as users that carry a laptop are expected to equally carry a SMD and as such when they travel between IPL locations they carry both devices with them. Table 6.4 shows that about 27% of all devices on IPL and 16% on ISEL are static, contributing for the 70% of the traces without movement. However, the size of the MobIPLity trace set is sufficient to attenuate this large proportion as more than 600000 traces for IPL can be found exhibiting movement.

The distribution of jump sizes (i.e. the distance travelled between waypoints) is depicted in Fig. 6.19. The irregular pattern observed at Fig. 6.19a, with knees at $100m$, $7000m$, $8000m$ and $10000m$ shows how the roaming of users among the multiple IPL campus impact the model. The smaller campus area of ISEL justifies the smaller travelled distances, which never exceed the $200m$.



(a) IPL/trace (b) ISEL/trace

Figure 6.19: Jump size (meters)



(a) IPL/trace (b) ISEL/trace

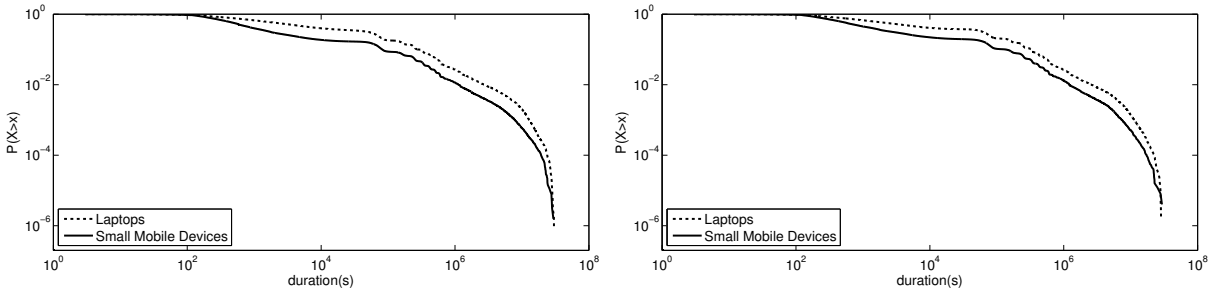
Figure 6.20: Pause times (s)

6.3.4 Pause Times

Figure 6.20 shows consistently briefer pause times for SMDs on both IPL and ISEL trace sets. This supports common knowledge of SMDs showing a higher mobility, what contrasts with the expected large pause times for laptops, typically operated by steady users. The longer tail on the plot for Laptops can be caused by devices that are kept at teachers offices, or at students dorms. The logarithmic scale of the graph hides the large difference between the maximum pause time for laptops (almost 8 days) and a maximum of 2 days for SMDs. The difference between these values is consistent on IPL and ISEL.

6.3.5 Disconnection Time

Figure 6.21 presents the CCDF for the time for which devices were disconnected, creating distinct traces. This metric was only obtained for devices that returned to the



(a) IPL/trace (b) ISEL/trace

Figure 6.21: Disconnection time (s)

network after a disconnection. The figure clearly shows the impact of the academic environment where the data was collected. The plot knees evidence a considerable number of disconnections of 12 hours, 2 days, 12 days, 2 months and 6 months. These periods represent either weekend/weekday periods, vacations and semesters. We also found that laptops have a higher probability of being disconnected frequently for periods of 90 minutes, which is the duration of classes. In contrast, the figure indicates that SMDs have traditionally smaller disconnection times, what is attributed to the power saving mechanisms available on these devices.

6.3.6 Scenario Analysis

This section uses some metrics discussed in the related work to evaluate the following 2 scenarios created using MobIPLity.

3 days This scenario puts together three traces of 2h each, extracted respectively from the 22nd of May, 18th of October and 6th of December 2012. The days and periods were individually selected to create a single $3600s$ scenario with the highest possible number of devices. This scenario was motivated by the objective of defining an environment as similar as possible to the one that can be more frequently found in the literature. Therefore, this scenario does not include devices whose trace terminated during the 2h period of the collection.

Disconnected The "disconnected" scenario further reflects the full potential of MobI-

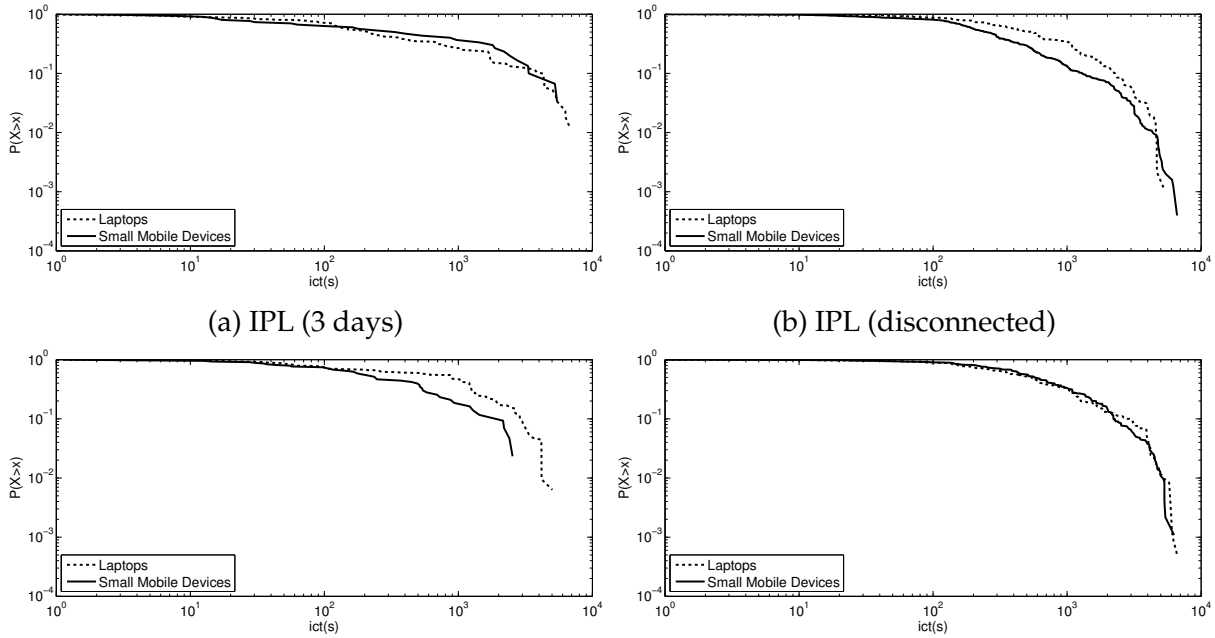
Table 6.5: Trace extraction options for the 3 days and disconnected scenarios

Parameter	3 days	Disconnected
Device' Type	Laptops/SMD	
Start/End Date	May 22, Oct 18, Dec 6, 2012	Oct 18, 2012
Points	2	
Duration	$7200s$	
Location	ISEL/IPL	
Axes	2D	
Warm Up	7200s	n.a.
Cool Off	7200s	n.a.
Enhanced trace	No	Yes
AP Range	$50m$	
Speed	$0.5ms^{-1}$	
Time	$120s$	

PLity by eliminating any constraints aimed to reproduce the conditions usually found in the literature. It was extracted from the 18th of October 2012, a date chosen because it presents the largest number of devices in a single $3600s$ trace. The disconnected scenario includes interrupted traces. Recall that interrupted traces include devices that turn off their radio during the period of the study.

Table 6.5 details the configuration parameters used in MobIPLity to produce both scenarios, according to the designation introduced in Table 6.2. The "disconnected" and the "3 day" scenarios share all the configuration parameters except for the option to include traces interrupted during the period. Expectations are that the differences observed between the two scenarios can give some hints on the impact of node disconnection.

Overall, this section considers 8 distinct scenarios, resulting from the combination of the two locations (ISEL/IPL), device type (Laptop/SMD) and Disconnected option. In the general case, all the scenarios consider 100 devices. The exception is for SMDs in the "3 days" scenario where only respectively 15, 20 and 18 devices for ISEL and 28,



(a) IPL (3 days)

(b) IPL (disconnected)

(c) ISEL (3 days)

(d) ISEL (disconnected)

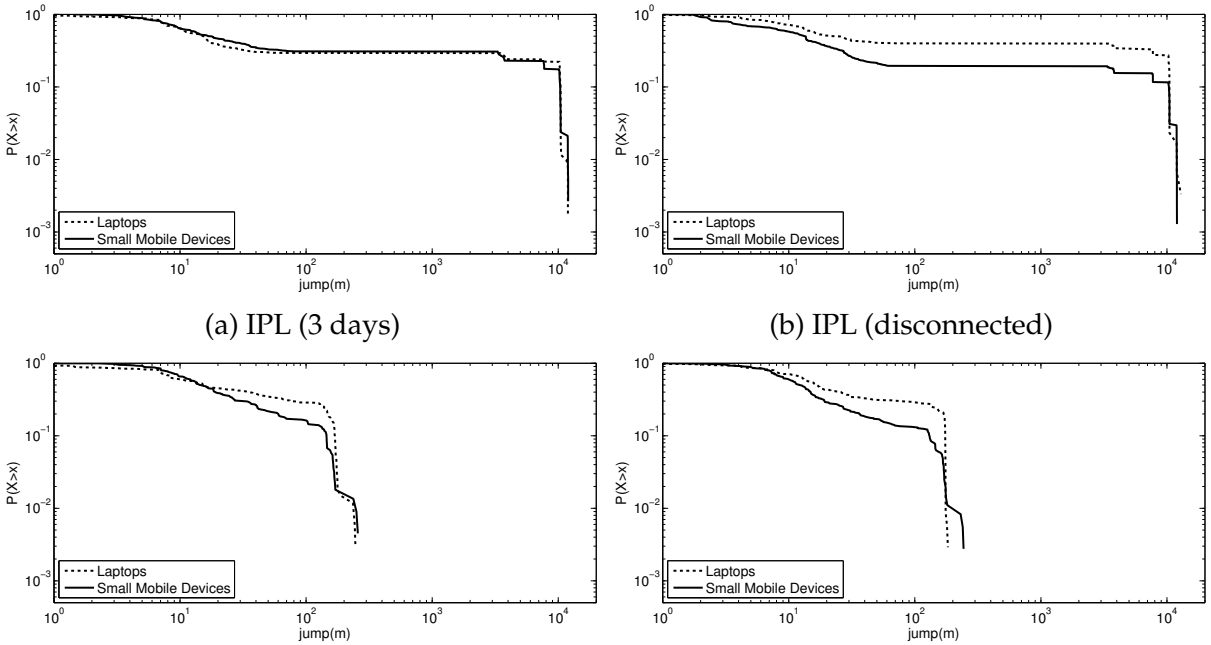
Figure 6.22: Inter-Contact Times (s)

43 and 45 devices for IPL could be found.

6.3.6.1 ICT

Figure 6.22 depicts the CCDF for the Inter-contact times (ICT) of the scenarios. It should be noted that the ICT metric only considers pairs of devices that become in touch a second time, ignoring all the cases where devices are in contact at most once. This supports the irregularity of the plots for the "3 day" scenarios, attributed to the small number of devices that remained connected during the 2h period.

Results show longer inter-contact times for devices that are geographically bounded to ISEL, what should be expected, considering the highest density of the network (in comparison with IPL) which increases the probability of the nodes to become in proximity more frequently. In general, plots suggest frequent re-connections among pairs of devices, with only 1% of them interrupted by more than $1000s$.



(a) IPL (3 days)

(b) IPL (disconnected)

(c) ISEL (3 days)

(d) ISEL (disconnected)

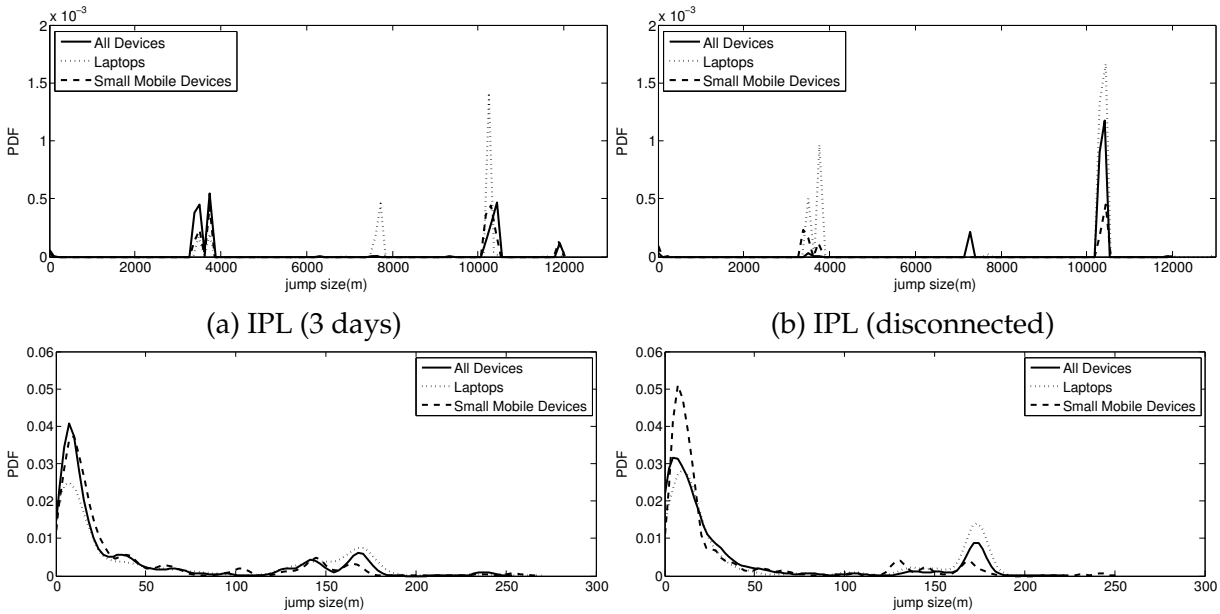
Figure 6.23: Jump Size (meters)

6.3.6.2 Jump Size

The distinctive dimensions of the IPL and ISEL campus become evident in the CCDF of the jump sizes depicted in Fig. 6.23. Results for IPL (Figs. 6.23a, 6.23b) exhibit a distinctive step pattern attributed to the distances between the different schools of the institution and to the need of some students and professors to commute between them.

A comparison between SMDs and Laptops shows that, in general, jump sizes of the former have a higher probability of being shorter than the latter. This can be attributed to the mobility of SMDs, which may connect to APs while being carred in the pocket of their users and which can be operated while the user is moving.

Figure 6.23a presents an interesting exception to the relation of the curves presented by laptops and SMDs given that laptops have a lower probabilities of moving throughout all the IPL. However, Fig. 6.23b contradicts the "3 day" results. Since the difference between both traces is restricted to the minimal speed of travel (which in Fig. 6.23a) must be above 0.5m/s), it is safe to assume that the abnormal behaviour is



(a) IPL (3 days)

(b) IPL (disconnected)

(c) ISEL (3 days)

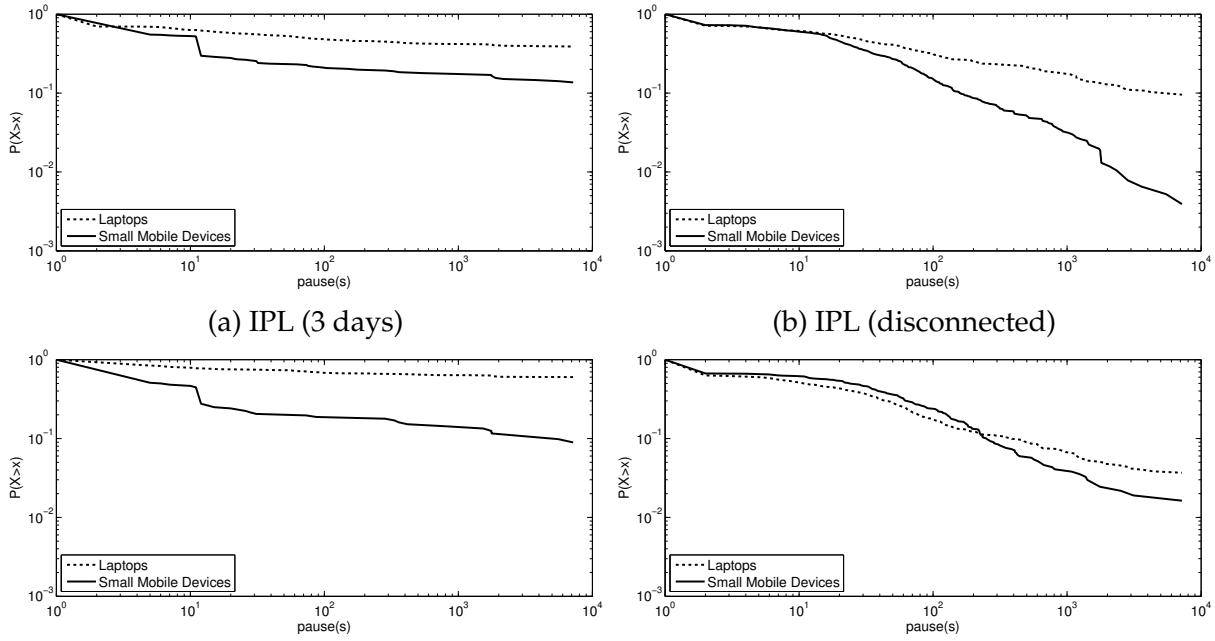
(d) ISEL (disconnected)

Figure 6.24: KDE of PDF distribution for jump sizes

due to the speed at which the devices travelled such long distances. In general, jump size results tend to support the claim that SMDs have a higher mobility, which produces larger traces passing through multiple APs, while laptops are disconnected and reconnected at a new location.

Jump sizes are tightly associated with the physical dispersion of IPL, which creates groups of users that either remain close on one school or travel between several. This issue has been previously identified, for example, in HCMM (Boldrini and Passarella, 2010) where scenario generation considers the possibility to set-up a number of groups and create bell shaped normal distributions.

Figure 6.24 shows the multiple kernel density estimations (KDE) of the Probability Distribution Function (PDF) of Jump sizes. The plots evidence the AP location distances, hinting at the limitations of positioning APs indoor, by producing jumps that relate to the distance between buildings or campuses.



(a) IPL (3 days)

(b) IPL (disconnected)

(c) ISEL (3 days)

(d) ISEL (disconnected)

Figure 6.25: Pause times (s)

6.3.6.3 Pause Times

The CCDF of pause times is presented on Fig. 6.25. It should be noted that in MobIPLity, pause times are particularly small as the methodology followed for trace definition tend to maintain a node in movement even if at a very small speed (Cf. Sec. 6.2.4.1). Therefore, a change in the methodology was applied. The results presented on the figure consider a device to be stopped if the distance between waypoints is less than 1 meter. Still, it is interesting to observe that SMDs have different pause time distributions with lower probabilities of having higher values, something that supports the mobility characteristics expected for SMDs.

Our results are in contrast with those presented in (Kim et al., 2006) where pause times were defined by detecting users walking at a low speed. The paper claims that pause times have a log-normal distribution, something that is not supported by our observations which show a power law distribution, supported by the Akaike test (see Table. 6.7).

Table 6.6: Inhomogeneity values

	All (σ)	Laptops (σ)	SMD (σ)
IPL-Oct 18th	0.79 (0.01)	0.9 (0.01)	0.77 (0)
ISEL-Oct 18th	0.59 (0.01)	0.62 (0.02)	0.68 (0.05)
IPL Disconnected	0.76 (0)	0.87 (0.01)	0.75 (0)
ISEL Disconnected	0.66 (0.03)	0.71 (0.01)	0.56 (0.03)
IPL RWP	0.4 (0.05)		
ISEL RWP	0.35 (0.06)		
IPL SLAW	0.63 (0.02)		
ISEL SLAW	0.52 (0.05)		

6.3.6.4 Inhomogeneity

Table 6.6 shows results for the inhomogeneity metric. Samples for this metric were obtained at 4 different times on the scenarios (at the beginning, 1/3, 2/3 and at the end of the scenario) of Oct 18th, the day with the most SMDs present on the network. Laptops show a higher inhomogeneity, indicating a larger concentration and irregular distribution for these devices. This is consistent with our expectations as it suggests that laptop users tend to be grouped, for example in classrooms or libraries. The lower value of inhomogeneity for SMDs confirms their pseudo-random deployment, where devices are connected while users move. The value of inhomogeneity for ISEL on the Oct 18th, presents a different pattern, possibly due to the low number of SMDs present on the network.

6.3.7 Comparison With Other Mobility Models

This section enabled the identification of a number of metrics that are "scenario agnostic". Trace duration, trace length, pause times, disconnection time and ICT are examples of metrics where differences observed between ISEL and IPL are minimal. On the opposite side, the evaluation also allowed to identify "scenario dependent metrics" of which trace speed, jump size and inhomogeneity are good examples. To facil-

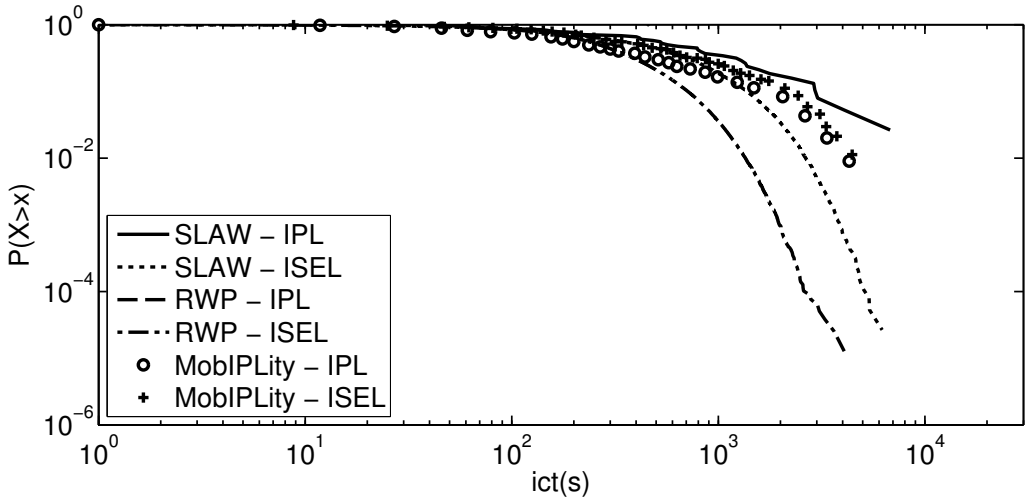


Figure 6.26: Comparison with ICT for SLAW and RWP

itate the comparison with trace-based mobility models, we compare MobIPLity with SLAW (Lee et al., 2009) and RWP using ICT (a scenario agnostic metric) and Inhomogeneity (a scenario dependent metric).

To compare ICTs, the setup and data presented in Sec. 6.3.6 was used. Scenarios that geographically emulate ISEL and IPL and with a similar number of devices were arranged for SLAW and the Random Waypoint (RWP). To better replicate the real conditions, the location of Access Points was passed to SLAW as hot-spots, enabling the creation of a model as accurate as possible. Figure 6.26 depicts and compares the CCDF results of ICTs for MobIPLity, RWP and SLAW. The figure shows that for IPL, RWP distributes the nodes so homogeneously that prevents generation of long ICTs, thus limiting the visibility of the data on the figure. Despite setting SLAW to emulate IPL on the number of access points, SLAW has limitations on representing the same ICTs as MobIPLity, which by itself presents similar ICTs for IPL and ISEL. Unfortunately, neither SLAW or RWP distinguish different device types although our results show that the device type plays a significant role on ICT.

The inhomogeneity metric, depicted in Table 6.6, was calculated for scenarios synthetically generated by SLAW and RWP that replicate the conditions found in IPL and ISEL (dimension, duration, devices and number of hotspots/access points). Results of our mobility records are similar to the ones obtained by SLAW, and as expected,

Table 6.7: Akaike test results

	SLAW	Dartmouth	MobIPLity		
	All	All	All	Laptops	SMDs
Duration	-	-	LN	W	LN
Dur./dev	-	-	G	G	P
Length	-	-	P	P	P
Len./dev	-	-	P	P	P
Speed	-	-	W	W	P
Speed/dev	-	E	P	P	P
Disconnection	-	-	GEV	LN	GEV
Pause	P	LN	P	P	P
Jump	P	-	P	P	P

(P: Pareto, LN: Log-Normal, G: Gamma, E: Exponential, W: Weibull, GEV: Extreme value)

diverge from the randomness found in RWP where the metric value is low.

To better understand the differences between metrics, mobility models and device types, the Akaike test was used to compare the fitness of ISEL traces for the complete year of 2012 to well-known distributions. ISEL traces were chosen due to their containment in a single campus, approximating the configurations found in the literature. Results presented in Table 6.7 were obtained using Matlab automatic fitting using all supported distributions and sorted by Akaike criteria. It is interesting to notice the difference between fitting results for different device types, where for the vast majority of SMD metrics Pareto is chosen as the best fit. This is in contrast with Laptops, with metrics being approximated by a bigger variety of distributions.

Results confirm our suspicions on the existence of different mobility models for these devices. The table shows the distributions used by SLAW and Dartmouth (Kim et al., 2006). The difference in chosen statistical models for pause times are evident. Pause time calculation methodology has impact on the fitting process. In (Kim et al., 2006), pause times were defined by detecting users walking at a low speed. The paper claims that pause times have a log-normal distribution, something that is not sup-

ported by our observations which show a power law distribution.

6.3.8 Discussion

In spite of the limitations dictated by the specific academic environment where the data was collected, MobIPLity allows the creation of distinct scenarios by considering the several schools of IPL and the distinct movement patterns that can be observed when considering Laptops, Small Mobile Devices and a combination of both. We claim that MobIPLity can be considered as a good starting point for the evaluation of a multitude of applications, including HTnT.

The recency of the dataset enabled the observation of the most recent pattern changes on mobility, that result from the increasing popularity of small dimension devices, effectively inducing a growth on user mobility. Interestingly, evaluation showed that power saving mechanisms that are standard on these devices reduce the possibility of spontaneous communication between peers and with the environment. This is an aspect that cannot be neglected by both mobility models and network simulators neither left for applications developers to address.

With the availability of the scenario generator we will proceed to the research on recurrence of contacts, in particular, on the capability to predict future repetitions of contacts between pairs of devices which can contribute to improve the dissemination of information among devices that are expected to be in range.

6.4 Recurrence of Contacts

In this work, contacts will be represented as temporal communities (TC) (Pietiläinen and Diot, 2012), defined as the set of devices connected simultaneously to the same AP. A TC exists as long as its membership does not change. The addition and/or removal of any member results in the creation of a new temporal community. The approach is oblivious to associations of devices to distinct APs with overlapping coverage and to

Table 6.8: Temporal communities observed in the dataset

Year	Max TC Size	TCs	Average TC Size
2005	33	370245	6.45
2006	43	1113630	8.16
2007	44	1309098	8.32
2008	66	1682684	9.37
2009	69	1700471	9.96
2010	74	1935039	11.08
2011	159	2775835	10.77
2012	121	5633825	10.08
2013	108	11366159	11.5

the repetition of TCs. TCs with the same membership are considered distinct if: $i)$ they occur in a distinct AP; or $ii)$ there is some interval between the two occurrences where an exactly equal TC did not exist. Table 6.8 summarizes the TCs counted using this approach.

Each TC with size n implicitly defines $\sum_{i=1}^{n} \binom{n}{i}$ *Temporal Sub-Communities* (TSCs), that result from all possible combinations of the members of the TC. It should be noted that TSCs include the special case where all the members of the TC are represented (i.e., the TC itself). Relevant for HTnT improvement is the evaluation of the repetitive occurrence of groups of devices, independently of its members being or not part of larger groups. Therefore, the work will focus mostly on the study of TSCs.

A multi-year analysis of the MobIPLity dataset shows a non-negligible variation of the number and size of the communities. Part of this variation can be attributed to the addition of Access Points (APs) to the network (cf. Fig 6.2), mostly motivated by the need to resolve localised network performance issues at the IPL. Such addition contributes to a decrease in the dimension of temporal communities as devices have more APs for association on the most frequently accessed locations.

Figure 6.27, which depicts the size of the biggest TC observed each day, clearly shows the impact of the academic environment on the network. In the figure it is

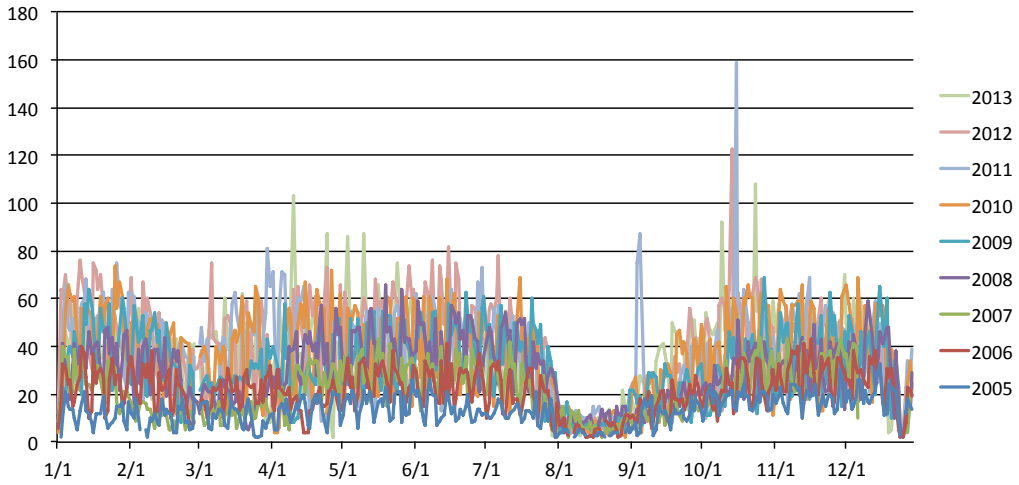


Figure 6.27: Max Community Size Per Day

possible to observe the reduced activity during the Winter (end of December), Summer (August) and Easter (March) breaks. The irregularity of the plots can also be attributed to weekends and to particular, non periodic events, such as conferences.

6.4.1 Temporal Communities

The accumulated number of Temporal Sub-Communities (TSCs) found in every day of 2012 is depicted in Fig. 6.28 as a Complementary Cumulative Distribution Function (CCDF). For clarity, the figure presents TSC sizes in steps of 6. It was observed that the lines of the TSC sizes that were omitted evolve similarly to those that are represented.

A first surprising effect observed in Fig. 6.28 is the peak of the number of TSCs at size 38. However, this is due to the methodology used for determining TSCs. Recall from Sec. 6.4, that the present work considers all possible combinations of elements of any TC as TSCs. In this case, each of the observed TCs of size 74 produces, by itself, $\binom{74}{38} \approx 1.7 \times 10^{21}$ TSCs with size 38. Still, the figure is illustrative of the potential number and size of the groups of devices within transmission range that can be found in academia. Notice for example that in 5% or more of the days of 2012 it is possible to find at least 10^{10} communities of size 62 and that 80% of the days had more than 100

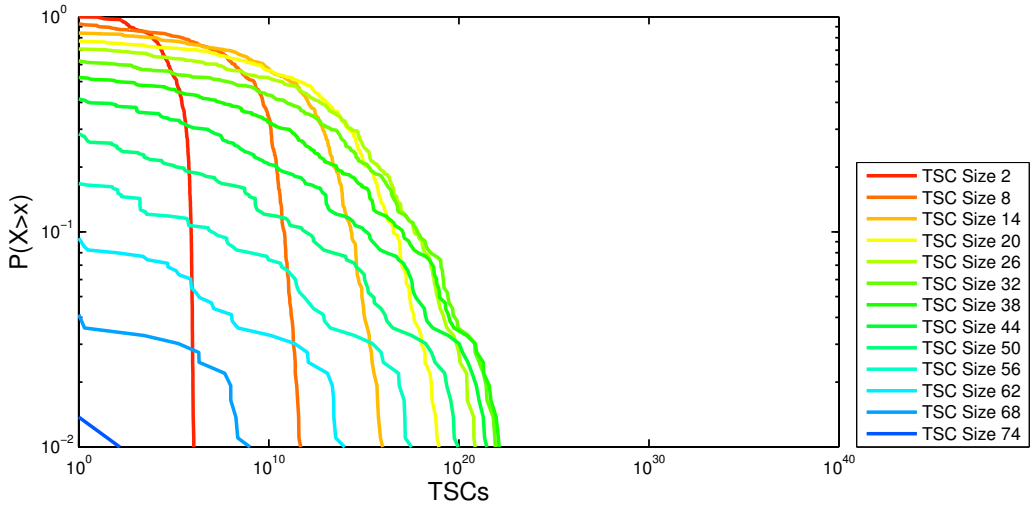


Figure 6.28: TSCs per day

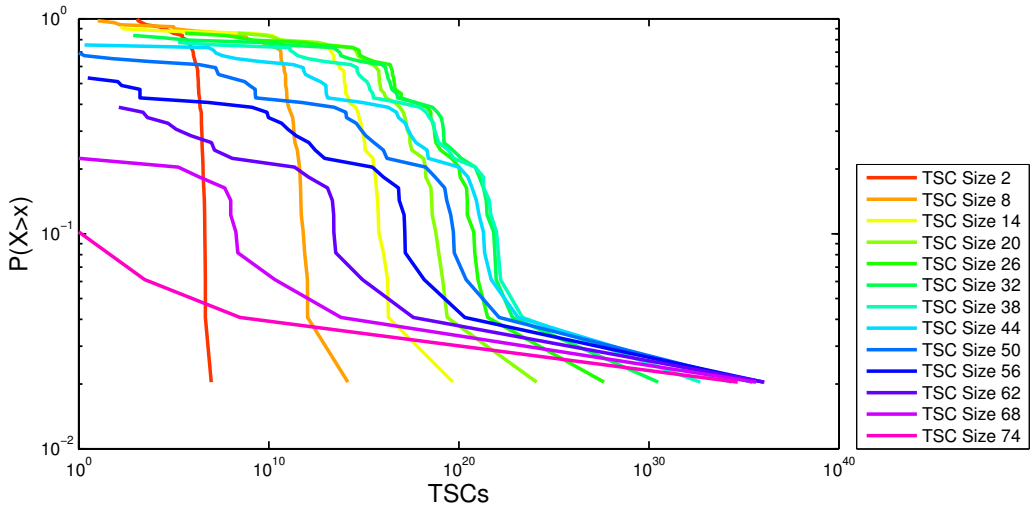


Figure 6.29: TSCs per week

TSCs with 14 devices.

Figure 6.29 revisits these results after grouping the TSCs in weeks, what removes the spurious effects of occasional TCs of very large size. Still, the plot denotes an interesting regularity, suggesting that TSCs of sizes up to 23 tend to occur in a large proportion of more than 90% of the weeks.

6.4.2 Temporal Patterns

In addition to presenting the number and size of all TSCs, the work evaluates the probability of recurrence of two and three consecutive hits of the same TSCs on 4 distinct temporal patterns. The **Consecutive Days (CD)** and the **Consecutive Week Day (CWD)** patterns use intervals of respectively 1 and 7 days. These patterns serve to investigate repetitions inspired by common student activities, like the daily attendance to school and the weekly attendance to classes.

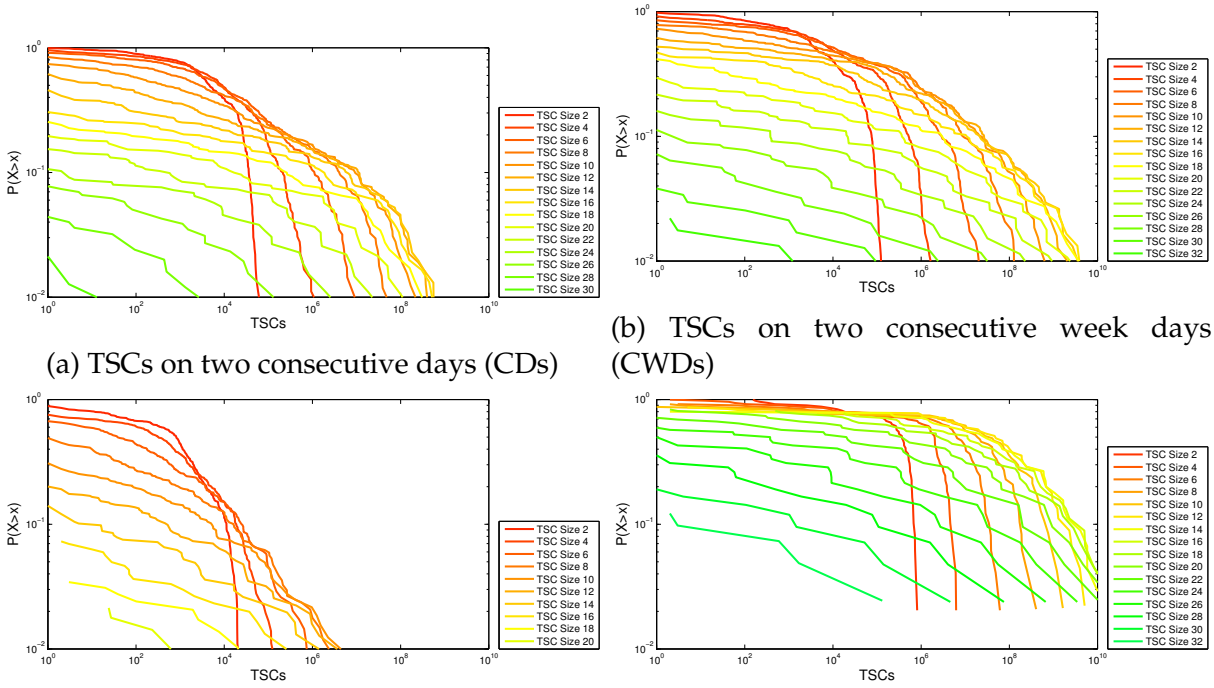
The **Consecutive Month Day (CMD)** and **Consecutive Week (CW)** use more irregular patterns. CMD seeks for repetitions in the same day number of consecutive months. CW in turns seeks repetitions in any weekday of consecutive weeks.

For clarity, and as an example, consider the observation of a TSC on July 18th, 2012 (Wed). A hit will be found if the same TSC is observed on July 19th for CD, July 25th for CWD, August 18th for CMD and on any day between the 22nd and the 28th of July (Sun-Sat) for CW.

These temporal patterns are affected by the negative impact of the calendar irregularities. No attempt to attenuate the effects of public holidays, weekends or school breaks has been made. This option was chosen to approximate the results from those found by some application using past experiences to estimate the probability of contact repetition.

6.4.3 Extraction of Contact Recurrence

Consider some TSC t observed in some instant i. The study on t's recurrence will proceed in two steps. First, it will measure the frequency with which t is observed a second time, respecting one of the temporal patterns defined in Sec 6.4. I.e., we will look for occurrences of t in instant i', knowing that the relationship between i and i' must necessarily respect one of the temporal patterns. The second step will evaluate the persistence of these occurrences. It estimates the probability of observing t in a



(a) TSCs on two consecutive days (CDs)

(b) TSCs on two consecutive week days (CWDs)

(c) TSCs on two consecutive month days (CMDs)

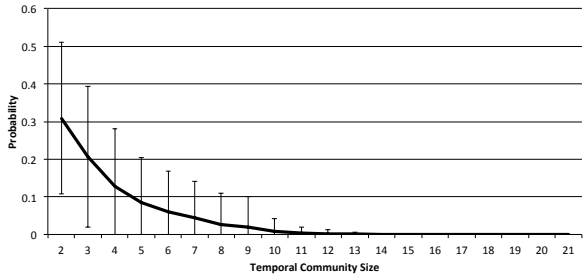
(d) TSCs on any day of two consecutive weeks (CWs)

Figure 6.30: Temporal patterns for two consecutive periods

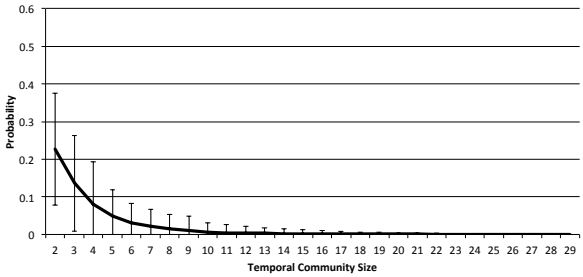
third instant i'', with the time interval between i' and i'' respecting the same temporal pattern that was found between i and i'. The analysis will focus in 2012, which presents a good trade-off between the manageability of the size of the dataset and its recency.

Figure 6.30 depicts the CCDF of the TSCs that were observed a second time satisfying each of the Temporal Patterns defined above. The figure clearly demonstrates that the selection of the temporal pattern has a strong impact on the results. The Consecutive Month Days (CMD) is the temporal pattern that performs poorly. This should be expected as it is hard to find any routines depending on the day of the month in the academic environment. In contrast, the CD and CWD temporal patterns, which reflect better the typical student schedule, perform reasonably well, specially for TSCs of size of 6 or less. In these cases, more than 90% of the days presented 100 or more TSCs which were equally observed in the previous instant of the temporal pattern.

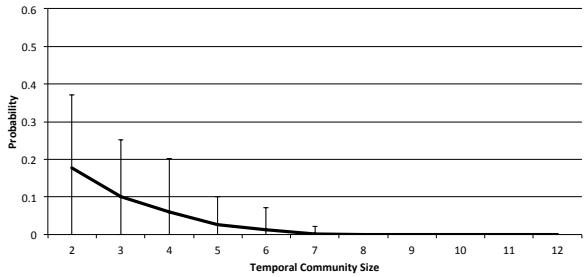
The best results are presented by the CW temporal pattern, where it was not hard to find 10000 communities of sizes 6 or less in 90% of the days. This is not a surpris-



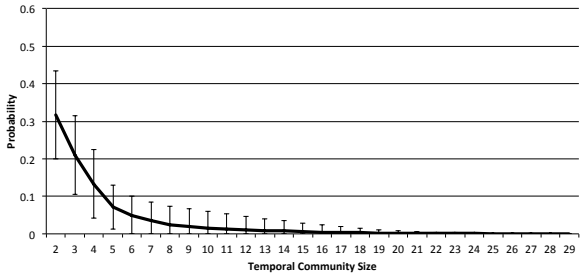
(a) Probability of finding the same TSC on three CDs



(b) Probability of finding the same TSC on three CWDs



(c) Probability of finding the same TSC on three CMDs



(d) Probability of finding the same TSC on three CWs

Figure 6.31: Probabilities for three consecutive periods

ing result considering the great flexibility of the constraints imposed by CW in comparison with CD and CWD. However, as it will be discussed later, the CW temporal pattern equally introduces a level of uncertainty that makes it unusable in prediction algorithms.

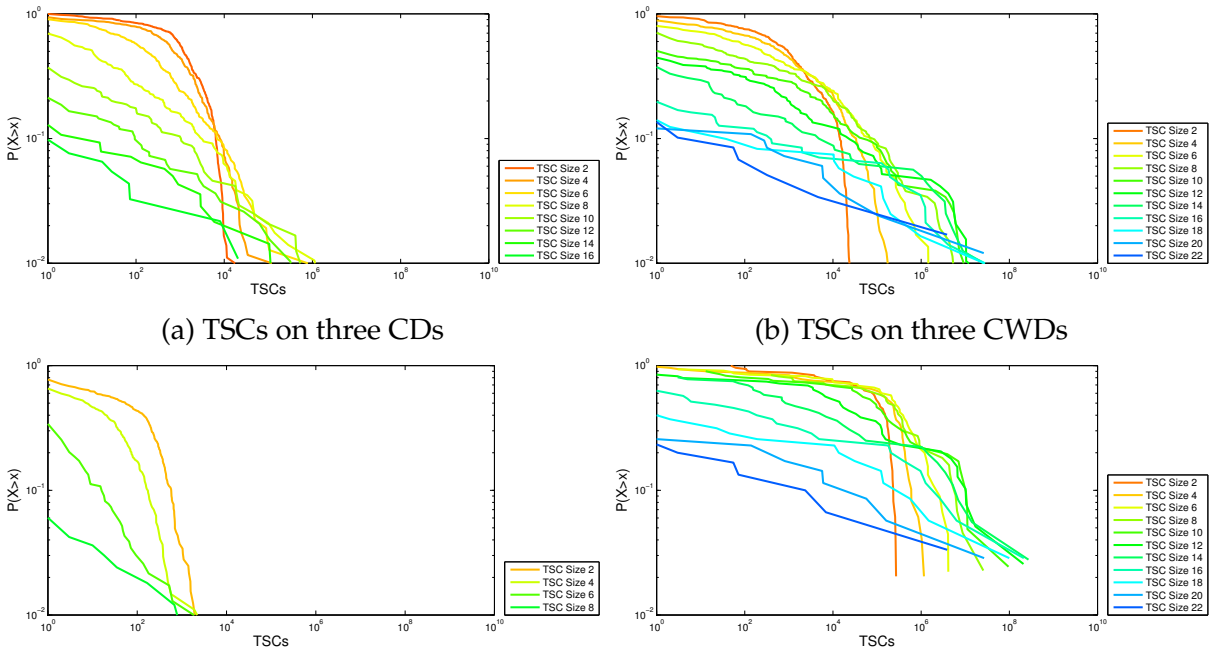
6.4.4 Predictability of multiple contacts

Fig. 6.31 shows the average and standard deviation of the proportion of TSCs that repeated in a third consecutive instant from those that were observed twice. Results contribute to decrease the relevance of the observations of large TSCs in two consecutive periods. Although it is frequent to find large TSCs, their membership tends to vary with time. Therefore, the occurrence of large TSCs can only be used by applications depending on ad hoc concentrations of users.

Concerning TSCs with small number of members, the results enabled the separation of the CD and CWD temporal patterns, with the daily one showing to be more

predictable. CD and the more relaxed CW temporal patterns are the only able to obtain probabilities of repetition above 10% for TSCs of up to 4 members and to show a surprisingly 30% for TSCs of size 2. This can be considered as a non-negligible probability of finding the same device on, respectively, the next two days and weeks. CW outperforms CD in the stability of the predictions, as it presents a smaller standard variation of the sample.

Figure 6.32 depicts these results using CCDFs of the absolute number of occurrences observed. As suggested by Fig. 6.31, TSCs with significant results are those with smaller membership sizes. It is also interesting to notice the distinct pattern exhibited by each temporal pattern. Still, it should be noticed that it is not hard to find a considerable number of TSCs satisfying the CD and CW criteria for 3 consecutive intervals. In the case of CW, in 90% of the days of 2012 it is possible to find 100 TSCs of 6 members that were observed on 3 consecutive weeks. In line with what was observed before, the CWD and CMD temporal patterns show disappointing results, in both the number of TSCs found and on the probability of their recurrence.



(a) TSCs on three CDs

(b) TSCs on three CWDs

(c) TSCs on three CMDs

(d) TSCs on any day of three CWs

Figure 6.32: Temporal patterns for three consecutive periods

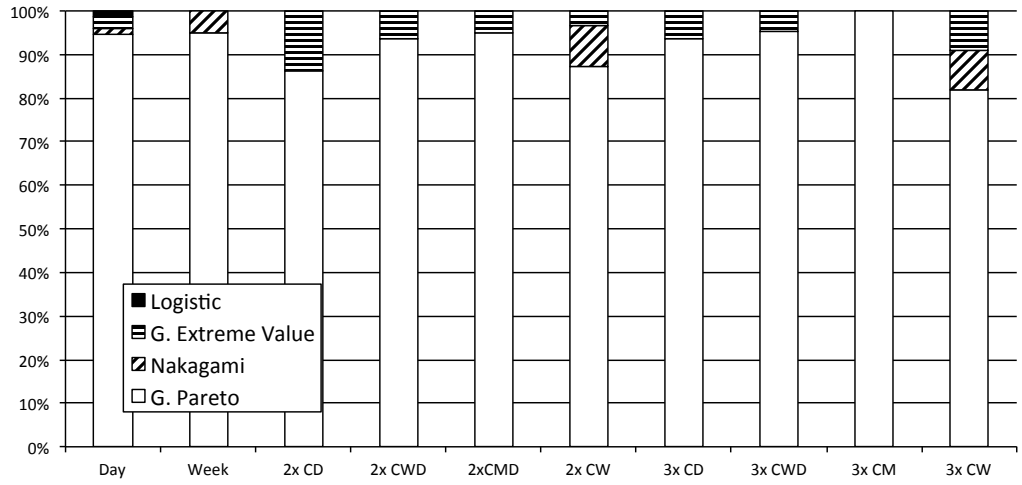


Figure 6.33: Distribution fitting

6.4.5 Probabilistic Model

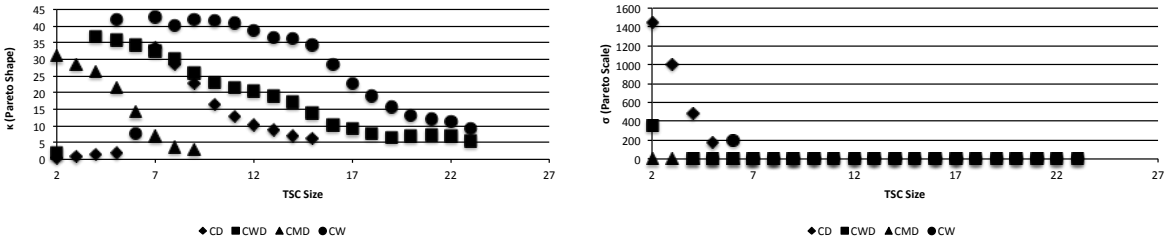
To model the temporal patterns observed in TSCs, the results discussed in Sec. 6.4.3 were fitted to statistical distributions using the Akaike information criterion on Matlab. Figure 6.33, which aggregates TSC sizes by distributions, shows that the Generalized Pareto distribution is the most adequate to model the behaviours observed our work. The use of the Pareto distribution is consistent with results found for modelling other aspects of human mobility, for example those detailed in (Gonzalez et al., 2008; Clauset et al., 2009; Karagiannis et al., 2010).

Interestingly, the CD, CWD and CMD temporal patterns exhibit an exception where only a single size of the TSCs is better represented by Generalized Extreme Value. As shown in Table 6.9, the sizes of the TSCs with an abnormal behaviour are distinct for each temporal pattern and no relation between the values could be found. Therefore, these cases are considered as an anomaly and the reminder of the text handles them indifferently from the remaining.

To make the knowledge obtained from the analysis of TSC useful for application developers, we analysed the relation between the parameters of the Generalized Pareto fittings, the dimension of the TSCs and the interval (in days) of the temporal patterns. The goal is to define a family of functions $f_{\beta}^{\alpha}(n)$ where $\alpha \in$ {CD,CWD,CMD} and $\beta \in$

Table 6.9: TSC size with exceptional distribution

Temporal Pattern	TSC Size
CD	6
CWD	3
CMD	2



(a) κ generalized Pareto parameter (b) σ generalized Pareto parameter

Figure 6.34: Generalized Pareto Parameters

$\{\kappa, \sigma\}$ that allow to rapidly define Generalized Pareto distributions for the estimation of repetition of temporal communities of size n. It should be noted that the remaining of this study omits the CW temporal pattern due to the impossibility in mapping it on a fixed time interval.

The approach followed consisted in finding suitable interpolations for the κ and σ values that shape the Generalized Pareto distribution. The parameters observed are depicted in Fig. 6.34 and evidence the abnormal behaviour at one specific TSC size for each temporal pattern.

Figure 6.35 shows σ after removing the values below the abnormal point addresses above.

The interpolation used the Matlab and eureqa applications, with the latter supporting the cases where Matlab was not able to provide an acceptable goodness of fit. The resulting family of functions is presented on Fig. 6.36 and considers the impact of the abnormal point. Functions for the CD temporal community are divided according to the dimension of the TSC pretended. For CWD, the Generalized Pareto is only applicable for TSC dimensions above the abnormal point.

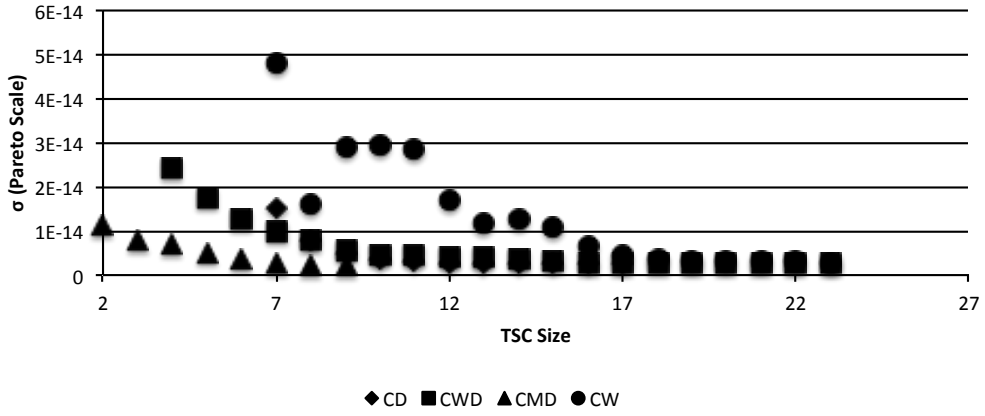


Figure 6.35: σ generalized Pareto parameter for values after the abnormal point

$$f_{\kappa}^{\text{CD}}(n) = \begin{cases} 0.06n^{2.1} + 0.2, & n < 6 \\ n + \frac{465}{n} - 40, & n > 6 \end{cases} \quad (6.1)$$

$$f_{\sigma}^{\text{CD}}(n) = \begin{cases} 859 + 450\cos(0.75n) + 548\sin(0.75n), & n < 6 \\ 10.095\text{e} - 12\exp(-0.8939n) - 15\exp(-0.0276n), & n > 6 \end{cases} \quad (6.2)$$

$$f_{\kappa}^{\text{CWD}}(n) = 39.71 + 0.007799n^{2.956} + 1.385\sin(2.579 + 0.8477n) - 0.2199n^2, n > 3 \quad (6.3)$$

$$f_{\sigma}^{\text{CWD}}(n) = 5.288\text{e} - 13\exp(-0.4151n) - 15 * \exp(-0.02243n), n > 3 \quad (6.4)$$

$$f_{\kappa}^{\text{CMD}}(n) = 40.58 - 4.239n - 2.042\log(n) * \sin(0.5316 + n), n \geq 2 \quad (6.5)$$

$$f_{\sigma}^{\text{CMD}}(n) = 0,273\text{e} - 14 - 16n^2 - 15n, n \geq 2 \quad (6.6)$$

Figure 6.36: Family of equations for modelling repetition of TSCs

6.4.6 Probabilities Function

A more practical application of these results can be obtained using probabilities, reproducing the approach discussed in Sec. 6.4.4. The extraction of a fitted function resulted in a simple Exponential curve with the form given by Eq. 6.7. Values for constants a and b depend of the temporal pattern and are given in Tab. 6.10.

$$PC_{tp}(n) = a_{tp} \times e^{b_{tp}n} \quad (6.7)$$

Table. 6.11 shows the values to be applied to Eq. 6.7 for obtaining the Standard Deviation metric. As discussed in Sec. 6.4.4 fitting is not as exact as for the Probability

Table 6.10: Probability function parameters

Temporal Pattern	a	b
CD	0.7167	-0.4204
CWD	0.6081	-0.4971
CMD	0.5947	-0.601
CW	0.7701	-0.4431

Table 6.11: Standard deviation function parameters

Temporal Pattern	a	b
CD	0.3335	-0.205
CWD	0.2375	-0.2216
CMD	0.4295	-0.357
CW	0.1499	-0.1391

Function. However, goodness of fit should be considered acceptable for all metrics of the equations presented.

6.5 Summary

The performance of the dissemination algorithm of HTnT could be improved with the capability to anticipate the number or the affiliation of the groups of devices to be found in the future. This work leverages on a large dataset of accesses to a number of eduroam access points at an academic institution to extract the complete set of temporal communities observed in 2012. Leveraging on these results, the chapter derives a statistical model that characterizes the different temporal community sizes assuming four distinct recurrence patterns that mirror likely schedules of the users.

The fitting of the observations showed that the recurrence of three temporal patterns can be modelled by generalized Pareto distributions, confirming the results already observed for the Inter-Contact Times.

This work presents a statistical model usable for prediction of neighbouring peer devices on a nearby future using the knowledge of the previous observed contacts and

temporal patterns.

This findings will be evaluated and applied on the modelling of future contacts for HTnT, not before a evaluation on the applicability of the knowledge obtained.

7

Application

The contact patterns observed in the dataset of the wifi access records from the eduroam network to create a contact prediction algorithm based on past observations. Expectations are that this algorithm contributes to improve the Hybrid Trust and Trade (HTnT) system of the Mobile Collaborative Cloudless Computing Environment (MC3E) by improving the reputation certificate dissemination algorithm used by the Central Trusted Entity (CTE), improving the detection of rogue devices. Contact patterns were extracted by converting the raw access records into bonnmotion (Aschenbruck et al., 2010) mobility scenarios. Bonnmotion was selected because of its frequent use by network simulators.

This chapter details the efforts and outcomes of this process, starting on Sec. 7.1 with a presentation of the web interface that make the data collected using the wifi access records publicly available. Sec. 7.2 reports on the contact prediction algorithm. Secion 7.3 details the evaluation of HTnT first by using a non-improved reputation dissemination algorithm and later by improving this algorithm with the prediction of contacts.

7.1 MobIPLity: Web Interface

To facilitate the dissemination of the traces, a web interface has been prepared and made publicly available at `http://edata.e.ipl.pt`. Expectation are that this dataset contributes for current research on human mobility, in particular, related with

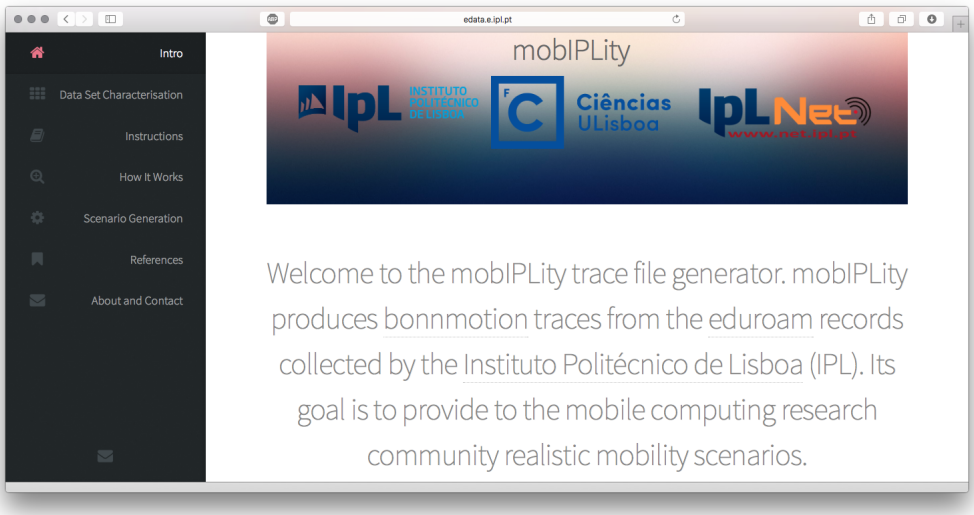


Figure 7.1: MobIPLity Webpage Screenshot

wireless networks. Traces are extracted on-demand from an E set stored at a local database and running the MobIPLity algorithm described in Sec. 6.2.4.1. The algorithm outputs traces in the format used by the bonnmotion mobility scenario generator and analysis tool (Aschenbruck et al., 2010). In compliance with bonnmotion trace format, time stamps are shifted, always starting of 0, independently of the effective date on the records. The initial waypoint for each trace is placed at a location predicted for each device at scenario starting time by MobIPLity. To further abstract effective user location, access points are consistently repositioned using a random factor.

The web interface (visible on Fig. 7.1) uses the parameters described in Tab. 6.2 (p. 84) and includes a set of suggested configuration options in order to accelerate the process of obtaining the bonnmotion scenario file.

7.1.1 Enforcement of User Privacy

A number of measures were put in place to protect the confidential nature of the data. Original records are kept at a secure location and are not disclosed in any circumstances. In compliance with the bonnmotion file format, the algorithm exclusively outputs (time,coordinates) pairs for the distinct devices. Therefore, no identification

that could be associated directly with a user or device is released. In addition, original data is obfuscated by: i) Repositioning waypoint coordinates in each new scenario generation while maintaining consistency among the location of the devices; and ii) starting all scenarios at time 0, without disclosing the offset between the requested start date and the effective beginning of the scenario.

To somewhat limit any judicious analysis of the data that could be crossed with information made available from other sources, all requests of scenarios will be moderated. Boundaries on the duration of the scenarios may also be applied.

7.2 Contact Prediction Algorithm

The capability to anticipate user contacts is valuable to a multitude of applications, which can be arranged according to the observer, in 3 distinct categories. In the *omniscient observer* category, a centralised server has access to the list of all contacts that have occurred in the past. An example of this application is HTnT, where a reputation server combines the contacts of all the service members to anticipate those that will occur in the future. The omniscient observer perspective is the one supporting the theoretical analysis from Sec. 6.4.5. In the *localised server* category, some external observer, for example an access point, creates a local perspective from the contacts he was able to observe, from his limited observation point. Finally, in the *peer view* category, each device anticipates future contacts exclusively from those where he has participated in the past. The peer view is the one where the information is more limited and therefore, where predictions are more challenging. This section reports on our efforts to create an algorithm capable to anticipate future contacts with a reasonable accuracy in the peer view perspective.

The algorithm leverages the distributions observed in the analysis of the eduroam dataset detailed in Sec. 6.2. It was designed to integrate with any application and accepts a target date and a list of the contacts observed in the past. Each contact is tagged with the date and duration of the contact and the peer ID. The algorithm outputs a

list of peers, ordered by the likelihood of finding it on the target day. Members of the output list are all the peers from the input contact list that satisfy any of the CD, CWD or CMD temporal patterns (described in Sec. 6.4.2) for the two previous consecutive instances and which, if observed on the target date, will result in a third consecutive occurrence of the pattern.

Peers are ranked according to the scoring function *score*, depicted in Eq. 7.1.

$$score = w_{CD} \times f_{CD}(d_{CD}) + w_{CWD} \times f_{CWD}(d_{CWD}) + w_{CMD} \times f_{CMD}(d_{CMD}) \tag{7.1}$$

where w_{tp} is the weight attributed to the temporal pattern and d_{CD}, d_{CWD}, d_{CMD} are the duration (in seconds) of the contact between the two nodes in the last event of the corresponding pattern. The *score* for each node is therefore dictated by an accumulation of partials from each temporal pattern where it was observed and given by:

$$f_{tp}(d) = PC_{tp}(2) \times CD_{tp}(d), \forall tp \in \{CD, CWD, CMD\} \tag{7.2}$$

where PC_{tp} is the probability function presented in Eq. 6.7 (p. 111).

The role of the CD_{tp} function is to convert the duration of the last contact in a weight. The function privileges longer contacts, mapping them on proportionally heavier weights. Two classes of functions depicted in Eq. 7.3 and 7.4 were experimented. Both assume that the contact duration is bounded between $60s$ and $86400s$ (one day), considered to be the interval representing social interactions between peers.

$$CD(d) = \frac{kd}{86400 + (k-1)d}, k \geq 1, d \geq 60 \tag{7.3}$$

$$CD(d) = \left(\frac{d}{86400}\right)^2, d \geq 60 \tag{7.4}$$

Functions differentiate by the direction of their curve. The family of functions of Eq. 7.3 increases the weight linearly with the duration when $k = 1$, increasing the weight of shorter contacts as k increases. In contrast, Eq. 7.4 tends to decrease the relevance of shorter contacts, increasing the weight more rapidly as the duration approaches 86400.

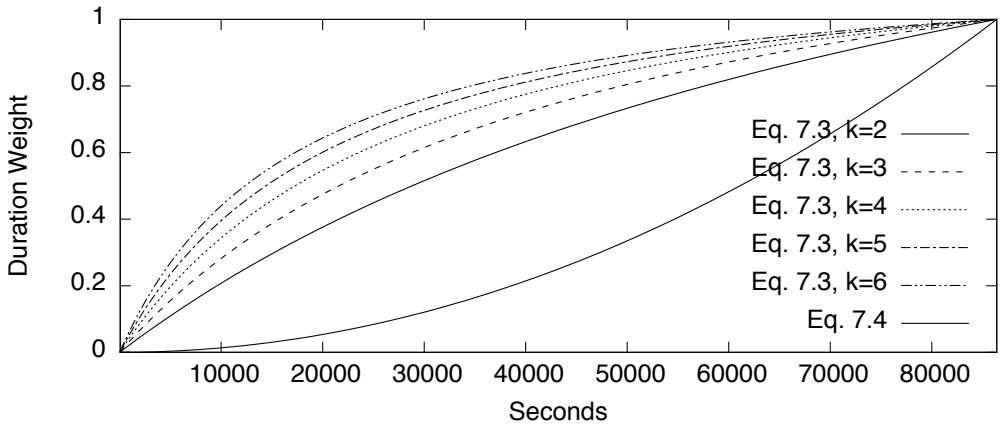


Figure 7.2: Equation 7.3 and Equation 7.4

Plots of both functions for representative values of k are depicted in Fig. 7.2.

Overall, this approach leverages from the probabilities of occurrence of a third repetition of a contact between a pair of nodes, which were found in the analytic study presented in Sec. 6.4.5, to derive a ranking algorithm. The algorithm uses the duration of the contacts and multiple temporal patterns between the same pair of nodes as tie breakers. Expectations are that these tie breakers correctly differentiate the likelihood of occurence of the third contact, providing to applications accurate estimates of upcoming contacts with other nodes.

7.2.1 Metrics

Evaluation was performed by running contact datasets against the algorithm and comparing its ordering with the contacts that have been actually observed. The capability of the algorithm to correctly order the expectations of contacts was evaluated using two metrics. Both metrics measure the number of hits (defined as a prediction of contact that has effectively occurred), although using different perspectives:

The *Rank of the First Miss* (RFM) metric returns the rank of the first failed prediction in the list. RFM is useful for application programmers as it indicates the number of highly reliable predictions of the list.

The second metric compares the proportion of hits on the 10, 25, 50, 75 and 100 percentiles of the list, identified in the remainder of the text respectively as p10, p25, p50, p75 and p100. Percentiles enable the evaluation of the quality of the ranking. Expectations are that the 100 percentile mirrors the analytic results discussed in Sec. 6.4.3. Therefore, the quality of the ranking will be evaluated by the increase in the proportion of hits in the lowest percentiles, which will confirm the capability of the algorithm to put hits at higher ranks.

7.2.2 Evaluation in MobIPLity

The ranking algorithm was experimented using a mobility scenario generated by MobIPLity with all devices that connected to the eduroam network on IPL during the year of 2013. Contact data was produced by configuring the *LinkDump* application of bonnmotion to extract the periods in which two peers were within a 50m range from each other for a minimum of $60s$. To prevent disturbance on the results due to the distinct patterns found on weekends, the original dataset was purged from the events occurring on Saturdays and Sundays.

It should be noted that the dataset used in the evaluation of the ranking algorithm is considerably distinct from the one used in Sec. 6.4.3 for the analytic evaluation of contacts. The later evaluated the 2012 dataset and defined a contact as the simultaneous association of two or more devices to the same access point, using RADIUS records. In this section, the 2013 dataset and a distinct methodology for defining contacts are used. In addition to exposing the ranking algorithm to a considerably distinct dataset from the one that inspired it, this approach enables the verification of the properties observed during the year 2012 are reproducible on a different year.

Parameters for the algorithm where experimentally tuned in order to obtain the best results for the metrics presented above. Table 7.1 depicts the experimental results for multiple variations of the CD functions with equal weights for w_{CD} and w_{CWD}. CMD was dismissed from this evaluation to reduce variants and due to a low impact

Table 7.1: Evaluation of contact duration functions

Equation	CD_{CD}	CD_{CWD}	rfm	p10
Eq. 7.3	$k = 4$	$k = 5$	3.61	0.40
	$k = 4$	$k = 6$	3.60	0.40
	$k = 2$	$k = 2$	3.58	0.40
	$k = 3$	$k = 6$	3.57	0.40
Eq. 7.4	–	–	3.16	0.39

on the final formula, given the lower values previously observed for $PC_{CMD}(2)$. Results evidence a minimal impact of the k constant when the function of Eq. 7.3 is used, and a considerably worse performance on the RFM metric when Eq. 7.4 is used. In practice, this result evidences a preference of the algorithm for a fast growing of the weight of the contact duration in the ranking. As a result, the reminder of the text presents results using Eq. 7.3 with $k = 4$ for all the CD_{tp} functions.

Table 7.2 and Table 7.3 shows the average and standard deviation of the metrics when different weights are used. These results average the rankings produced for all devices and days, provided that the ranking contained 20 or more devices. The tables show some encouraging results. In particular, that the algorithm can correctly rank on average the first 3.6 devices and that 40% of the highest 10% ranked devices have been found as predicted.

The contribution of the algorithm becomes more evident by noticing that a random sort of the list would equally distribute the 31% of the devices on the list that were effectively observed (p100) by all the percentiles. Figure 7.3 further emphasis this result by evidencing the 28% performance gain of p10 over p100. A combined analysis of the figure and of the Table 7.3 highlights the distinct contribution of each of the temporal patterns to the algorithm, with the participation of the CMD or the use of individual patterns consistently presenting worse results than a 50%,50% combination of weights of CD and CWD.

In the elaboration of the results above, a ranking list is always prepared, independently of an effective connection of the device to the network being observed. How-

Table 7.2: Totals and RFM results (average per day and standard deviation)

w_{CD}	w_{CWD}	w_{CMD}	Number of ranks	rfm (σ)
50	50	0	56892626	3.609 (6.75)
50	0	50	38423918	3.352 (7.13)
33	33	33	65604286	3.326 (6.30)
0	50	50	40827514	2.690 (3.81)

Table 7.3: Percentile results (average per day and standard deviation)

w_{CD}	w_{CWD}	w_{CMD}	p10 (σ)	p25 (σ)	p50 (σ)	p75 (σ)	p100 (σ)
50	50	0	0.402 (0.38)	0.377 (0.35)	0.352 (0.32)	0.333 (0.30)	0.313 (0.29)
50	0	50	0.361 (0.37)	0.340 (0.34)	0.317 (0.32)	0.301 (0.30)	0.283 (0.28)
33	33	33	0.363 (0.37)	0.340 (0.34)	0.316 (0.32)	0.297 (0.30)	0.279 (0.28)
0	50	50	0.360 (0.37)	0.342 (0.35)	0.322 (0.33)	0.306 (0.31)	0.288 (0.29)

ever, numerous cases were found where some devices did not connect to any another device in one complete day, although the algorithm predicted some connections. As can be confirmed by Tab. 7.4 and Tab. 7.5, this cannot be considered a negligible aspect. The table presents the same metrics after excluding these lists. Not surprisingly, lists become much more accurate, with 100 percentile approaching an average of 50%. I.e., on average 50% of the devices predicted by the algorithm are effectively found. More demanding metrics, in particular RFM (visible on Tab. 7.4) and p10 are in line with the improvement of p100: on average, the first 5 devices of each list are effectively found as predicted as well as more than 63% of the percentile 10 of each ranking list.

Table 7.6 discloses a final analysis of these results by presenting the metrics per day of the week. It is interesting to notice that the performance of the algorithm is not uniform across all the weekdays. The ranking algorithm presents better results for Tuesday, Wednesday and Thursday. The worst results of Mondays can be attributed to the weekend discontinuity impact on the CD temporal pattern. Surprisingly, Fridays present the worst performing results, although no evident explanation could be found.

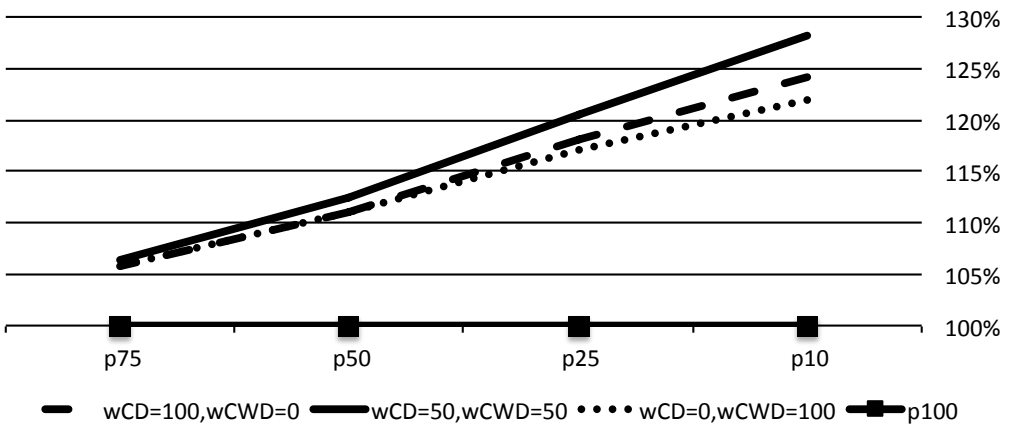


Figure 7.3: Improvement observed

Table 7.4: Totals and RFM results excluding not connected days (average per day and standard deviation)

w_{CD}	w_{CWD}	w_{CMD}	Rank Totals	rfm (σ)
50	50	0	39620508	5.122 (8.11)
33	33	33	43290708	4.967 (7.82)
50	0	50	25271814	4.871 (8.82)
0	50	50	25861694	3.917 (4.64)

7.2.3 Evaluation with Taxi Traces

To understand the applicability of the algorithm in other scenarios, the same experiment was applied to a dataset containing 1 month GPS traces of 320 taxis in Rome (Bracciale et al., 2014). The dataset was sanitized to include only positions in the metropolitan area of Rome, and to mark as off-line the taxis not reporting their position for an interval of $120s$ or more.

Table 7.7 and Table 7.8 show results of the metrics presented above exhibited by the algorithm in the taxis scenario. Unfortunately, the smaller and shorter trace prevented experiments with the CMD temporal pattern and forced to a reduction of the minimum size of the rankings to be considered in the evaluation from 20 to 5. Surprisingly, p10 metric shows values comparable to the ones obtained from MobIPLity, for the same configuration parameters. The differences in the RFM can be attributed to the smaller dimension of the dataset, which necessarily reduces the ranking list and,

Table 7.5: Percentile results excluding not connected days (average per day and standard deviation)

w_{CD}	w_{CWD}	w_{CMD}	p10 (σ)	p25 (σ)	p50 (σ)	p75 (σ)	p100 (σ)
50	50	0	0.635 (0.28)	0.596 (0.24)	0.557 (0.22)	0.526 (0.21)	0.495 (0.20)
33	33	33	0.620 (0.28)	0.579 (0.25)	0.538 (0.23)	0.507 (0.21)	0.476 (0.20)
50	0	50	0.593 (0.29)	0.559 (0.26)	0.522 (0.24)	0.495 (0.23)	0.466 (0.21)
0	50	50	0.621 (0.28)	0.591 (0.25)	0.556 (0.23)	0.528 (0.22)	0.498 (0.21)

Table 7.6: Per day of the week metrics for setup with the highest improvement (wCD=50,wCWD=50)

	rfm	p10	p25	p50	p75	p100
Monday	3.15 (4.62)	0.4 (0.38)	0.38 (0.35)	0.35 (0.32)	0.34 (0.31)	0.32 (0.29)
Tuesday	4.13 (7.57)	0.45 (0.38)	0.42 (0.35)	0.39 (0.33)	0.37 (0.31)	0.35 (0.3)
Wednesday	3.97 (8.26)	0.41 (0.38)	0.39 (0.34)	0.36 (0.32)	0.34 (0.3)	0.32 (0.28)
Thursday	3.67 (6.65)	0.41 (0.38)	0.38 (0.35)	0.36 (0.32)	0.34 (0.31)	0.32 (0.29)
Friday	3.04 (5.65)	0.34 (0.36)	0.32 (0.33)	0.29 (0.3)	0.27 (0.28)	0.26 (0.26)

proportionally impacts RFM. The difference between p10 and p100 loses significance, preserving a maximum gain of 11%.

Table 7.9 show the outcome of the per day of the week analysis. One can observe that in this dataset, Monday is the worst performing day of the ranking algorithm. This result is attributed to the discarding of weekends that was kept from the MobIPLity analysis in an attempt to keep the comparison fair. However, the social constraints that encouraged the introduction of the exception for MobIPLity have no significance in a taxis scenario, where devices are expected to operate on all days of the week. To the extent of our knowledge, this was the unique characteristic of the algorithm which did not adapt to both scenarios.

7.2.4 Discussion

In contrast with our expectations, differences in results between the consecutive day (CD) and consecutive week day (CWD) temporal patterns appear to be orthogonal to the environment. As an example, one could consider that the CD temporal pattern

Table 7.7: Taxis in Rome trace totals and RFM results

w_{CD}	w_{CWD}	Rank Totals	rfm (σ)
50	50	3487	1.884 (1.39)

Table 7.8: Taxis in Rome trace percentile results

w_{CD}	w_{CWD}	p10 (σ)	p25 (σ)	p50 (σ)	p75 (σ)	p100 (σ)
50	50	0.426 (0.49)	0.409 (0.43)	0.380 (0.33)	0.340 (0.28)	0.312 (0.25)

better represents faculty (with a daily schedule), and CWD would better represent students that meet in classrooms following a weekly schedule. Surprisingly, our study shows that the Taxis in Rome trace present comparable results. Therefore, we can consider that, to some extent, this supports the usage of our algorithm and probability modelling on multiple environments.

Results suggest that performance could be improved by considering weekdays in the definition of the ranking algorithm. However, this claim must be supported by additional experiments in other traces.

7.3 Reputation System Evaluation

In order to evaluate the Hybrid Trust and Trade (HTnT) system of the Mobile Collaborative Cloudless Computing Environment (MC3E), described in Sec. 5.2, we start by using a synthetic mobility model, reduced duration, number of devices and other commonly used parameters, to test the ability of HTnT detecting rogue devices with different profiles by disseminating reputation certificates using global and local trust. In this evaluation we use a simple probabilistic dissemination algorithm, where the CTE disseminates lists where devices with the lowest reputation have a higher probability of being included. The device with the lowest reputation is sent to all other devices, the second lowest has a probability of 0.5 of being sent to a device, the third lowest a probability of 0.33 and so on.

Table 7.9: Per day of the week metrics for taxis in Rome, setup with the highest improvement (wCD=50,wCWD=50)

	rfm	p10	p25	p50	p75	p100
Monday	1.27 (0.61)	0.19 (0.39)	0.2 (0.33)	0.18 (0.21)	0.16 (0.17)	0.15 (0.16)
Tuesday	1.87 (1.18)	0.48 (0.5)	0.42 (0.42)	0.41 (0.3)	0.39 (0.28)	0.35 (0.21)
Wednesday	1.95 (1.41)	0.44 (0.5)	0.41 (0.45)	0.4 (0.33)	0.34 (0.28)	0.31 (0.25)
Thursday	2.03 (1.5)	0.45 (0.5)	0.43 (0.45)	0.4 (0.35)	0.35 (0.3)	0.32 (0.25)
Friday	2.16 (1.58)	0.57 (0.5)	0.54 (0.41)	0.49 (0.3)	0.46 (0.27)	0.42 (0.24)

After the evaluation of the HTnT protocol we proceed with the evaluation of the dissemination algorithm, using as input a list of predicted contacts and list of devices in range extracted from bonnmotion LinkDump application. HTnT will benefit from the knowledge of the contact prediction algorithm to choose which reputation certificates should the CTE send to devices.

7.3.1 Synthetic Mobility Model Evaluation

To evaluate the capability of HTnT to classify the behaviour of participants in MC3E, simulations using the OMNeT++ network simulator with the INET framework were performed. HTnT was implemented as a standard UDP application, on a network stack of a 802.11 ad hoc mobile device with a 11Mbps link.

Devices are configured to move on a 2D open area of $9km^2$ and have a 250m transmission radius. Mobility parameters change according to a Gaussian distribution with a 2s mean and a 0.5s standard deviation. Every change in mobility implies an angle change of a maximum of 30 degrees and a change in speed using a Gaussian distribution with $20ms^{-1}$ mean and a standard deviation of $8ms^{-1}$. Table 7.10 presents the default settings for the simulation. This configuration represents numerous devices moving in the streets of a metropolitan area using the mass mobility model (Perkins and Wang, 1999).

To simulate the periods where the Internet connection is offline, the CTE is modeled with a probability of being *online* or *offline* for each device. During the contacts with the

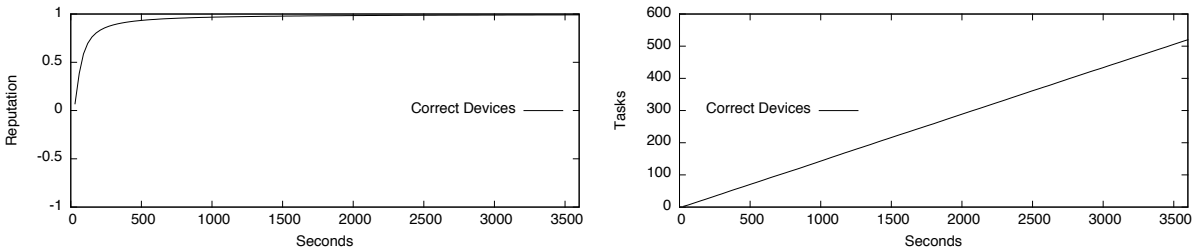
Table 7.10: Simulation Parameters

Network	Num. of correct devices	100
	Num. of rogue devices	10
	Task cost	10
CTE	Initial number of TwinCoins for each device	5000
	Reputation certificates cache size	10
	CTE transaction commission	0.02
	Device min. reputation for credit redistribution	0.5
Devices	α for GR_u	0.7
	Initial TwinCoins withdrawn	100
	Initial reputation	0
	Reputation certificate trust threshold	0
	Device ignore reputation threshold	-0.5
	Outdated reputation certificate threshold	5s
Simulation	Duration	3600s
	Simulated area	9km^2

central trusted entity (CTE), each device updates its reputation certificate and receives the reputation information of 10 other devices.

The simulation proceeds in cycles. Cycles have a periodicity of 0.5s. On each cycle, a device selects one of five actions with a probability indicated between (): i) withdraw (0.1); ii) deposit and report failed transactions (0.1); iii) change pseudonym (0.1); iv) request a task (0.2); v) remain inactive (0.5). Devices always volunteer to execute a task requested by another peer. Simulations were configured so that nodes with $GR_u < -0.5$ are never selected by their peers. Reputation certificates with $GR_u < 0.5$ cause a local reputation quest. In this process, devices will always assume the lowest reputation information value learnt to be the correct one.

Evaluation is focused on two metrics. The *reputation* of each node is measured by the value on the signed certificates delivered by the CTE on withdraw, deposit and pseudonym change operations. This metric shows how the system is capable to detect the misbehaviour of the users. The capability of each node to continue to participate on the MC3E is measured by the *number of tasks* sent. This metric shows the average of the total number of tasks requested by each node, counted when the first coin is



(a) Reputation (b) Sent tasks

Figure 7.4: Baseline simulation, no rogue devices, CTE online

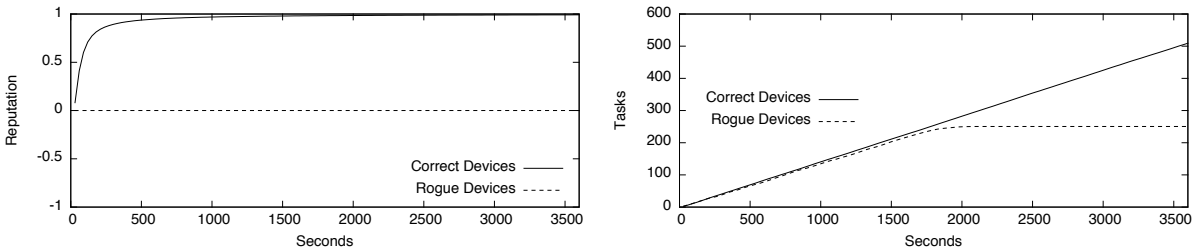
delivered to the sharer. For these two metrics, plots present the average of samples obtained with $30s$ intervals for each device.

7.3.2 Reference Simulation

To confirm that the system exhibits the desired properties in ideal conditions, a preliminary baseline simulation was performed, without rogue devices or pseudonym changes and uniformly distributing the probability for each of the remaining 4 actions. Results have shown that devices will not be prevented from requesting tasks as in these conditions' reputation increases rapidly and remains stable, as seen on Fig. 7.4. This was equally confirmed by the steady increase in the number of tasks submitted.

7.3.3 Resilience to Selfish Behaviours

To evaluate the performance of the system in the presence of rogue devices, different misbehaviour profiles have been defined. Simulations assume that all the attacks are detected, allowing us to focus on the effects of the attacks on the MC3E for correct devices. Unless a different value is indicated, simulations where configured with 10% of rogue devices.



(a) Reputation (b) Sent tasks

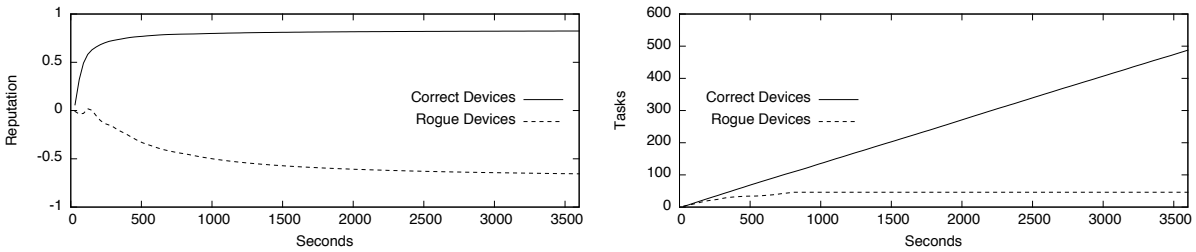
Figure 7.5: Threat model 1 simulation

7.3.3.1 Threat Model 1: Selfish Device

In this model the rogue devices do not accept tasks from other devices. The model validates the incentive proprieties of HTnT, addressing requirement R1 of Sec. 5.1. Figure 7.5 shows that the reputation of selfish devices remains neutral, what is expected given that the system only increases the reputation of sharers. However, the participation of selfish nodes on the system is limited by the initial credits provided to the nodes. As this credit is exhausted, so is their capability to submit tasks. This pattern is not observable on the correct devices, which continue to benefit of the MC3E.

7.3.3.2 Threat Model 2: Virtual Currency Protocol Attack

In this model, rogue devices start by executing a task request operation. However, after receiving the results from the sharer device, the rogue devices don't deliver the second coin of the TwinCoin. This behaviour prevents the sharer from receiving the corresponding TwinCoins amount, a problem that is reported on the next interaction of the sharer with the CTE. Figure 7.6 this behaviour becomes eventually evident for altruistic devices. After some transactions, the spreading of the reputation allows the remaining devices to stop accepting task proposals from the rogue devices. The virtual currency that the rogue devices have in their purse to execute tasks stops from being accepted, making them unable to send tasks to the MC3E, an aspect that can be observed in Fig. 7.6b. It should be noted that the pattern is not affected by the reputation gains obtained by providing services nor by pseudonym changes, given the



(a) Reputation (b) Sent tasks

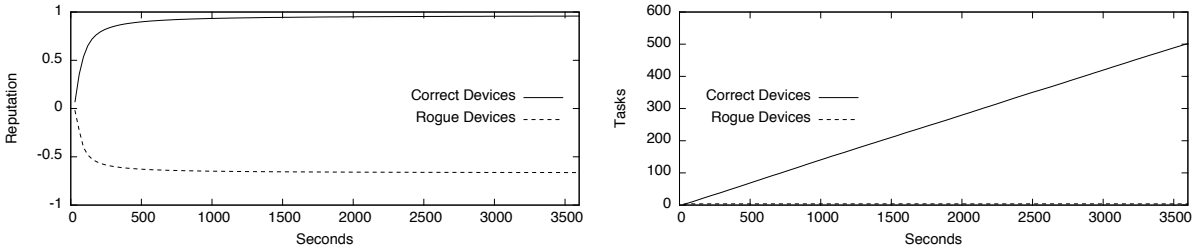
Figure 7.6: Threat model 2 simulation

impossibility to present a recent certificate with an acceptable reputation. During this experience it was observed that rogue devices have more virtual currency than the rest of the devices, although they are unable to use it for requesting services.

A collateral effect of this attack is the reduction of the total amount of TwinCoins in circulation, that are discarded by the CTE on each complaint. To address this issue, the CTE re-injects the lost virtual currency in the system, depositing it in the account of all devices whose $GR_u > 0.5$. Figure 7.6 equally shows a more irregular pattern in the increase of the reputation of correct devices, which is due to the penalty applied to complainers. However, as expected, this penalty is not sufficient to prevent correct devices from continuing to benefit of the system.

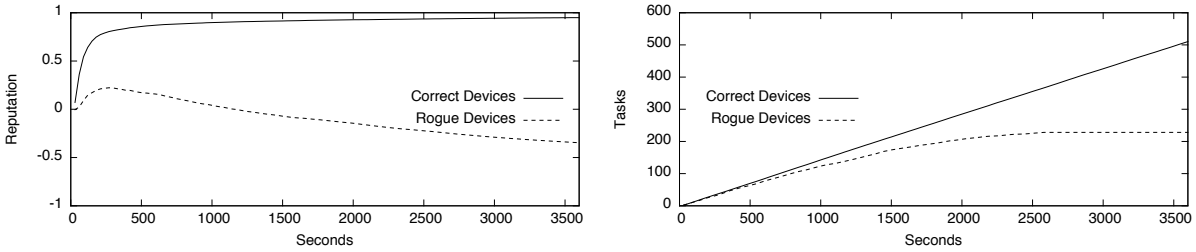
7.3.3.3 Threat Model 3: Virtual Currency Exhaustion Attack

In this set of experiments rogue devices accept the first coin but do not provide the service, what is later detected by the consumers. Results depicted in Fig. 7.7 show that the system rapidly identifies and penalises the misbehaving devices. It is interesting to notice that the reputation of the rogue devices decay at a much faster pace than in the Threat Model 2. This is due to the fact that in this attack, rogue devices do not increase their reputation by providing services.



(a) Reputation (b) Sent tasks

Figure 7.7: Threat model 3 simulation



(a) Reputation (b) Sent tasks

Figure 7.8: Threat model 4 simulation

7.3.3.4 Threat Model 4: Smart Rogue Device

In this set of experiments, a device executes a malicious action (uniformly chosen from the ones executed at the Threat Models 2 or 3) on his first cycle with each pseudonym and behaves correctly until the next change of pseudonym. To increase the number of malicious actions, in this scenario, rogue devices request a pseudonym change with probability 0.3.

Results depicted in Fig. 7.8a show that the reputation of rogue devices decays at a much slower pace than in the remaining threat models. This results from the number of correct actions that the nodes perform and suggests that it is possible for devices to play judiciously with their reputation in order to obtain some benefits from the system. However, Fig. 7.8b show that the number of tasks accepted by the peers falls, indicating that the system is still able to apply some penalties.

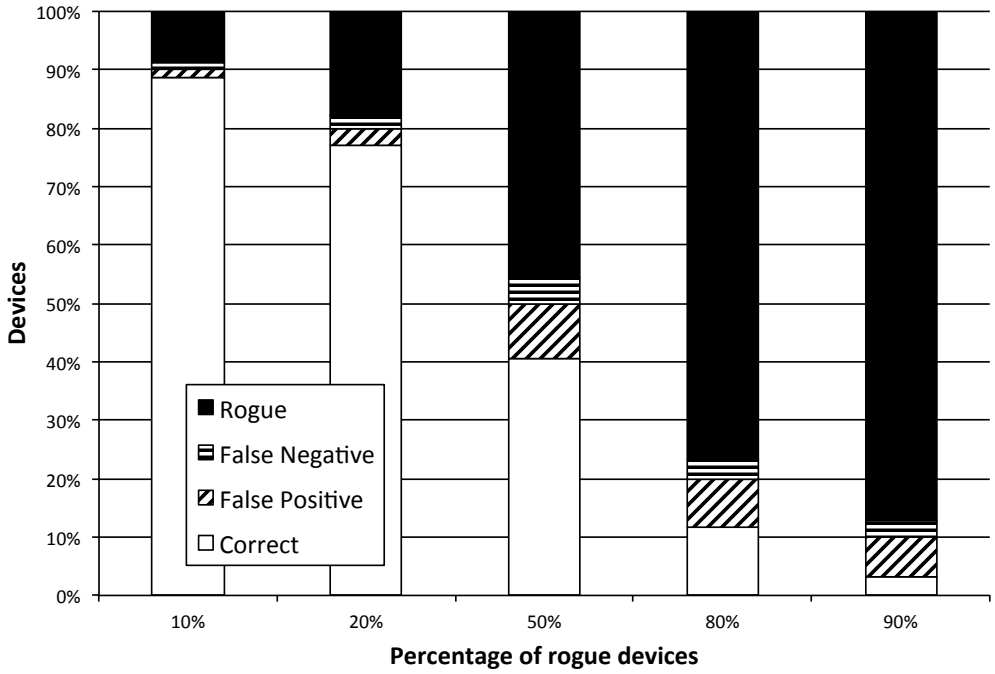


Figure 7.9: Resilience to rogue devices

7.3.4 Scalability

Results above confirm the resilience of a hybrid trust and trade system to attacks from a small (10%) number of participants. To evaluate the system tolerance to the presence of a high number of rogue devices, simulations with 10%, 20%, 50%, 80% and 90% of nodes executing the Threat Model 4 were performed. At the end of each simulation the reputation of each device recorded on the CTE was compared with their predefined behaviour and arranged in 1 of 4 categories. *Correct* and *Rogue* are those devices programmed respectively to behave correctly and roughly and having a corresponding positive or negative reputation. *False negatives* (resp. *positives*) are rogue (resp. correct) devices with positive (resp. negative) reputation.

The average of the results of five simulations for each proportion of rogue devices is depicted in Fig. 7.9. The results show that the service copes well with proportions of up to 50% of rogue devices after which its accuracy begins to degrade. A result that may be considered acceptable, given the complexity of programming mobile devices to exhibit the pattern of the Threat Model 4. In addition, it should be noted that this

Table 7.11: Simulation Parameters for HTnTv2

Network	Task cost	10
CTE	Initial number of TwinCoins for each device	1000
	Reputation certificates cache size	100
	Threshold for reputation to be sent to devices	0
Devices	α for GR_u	0.7
	Initial TwinCoins withdrawn	100
	Initial reputation	0
	Reputation certificate trust threshold	0.5
	Device ignore reputation threshold	-0.5
	Transaction duration	10s

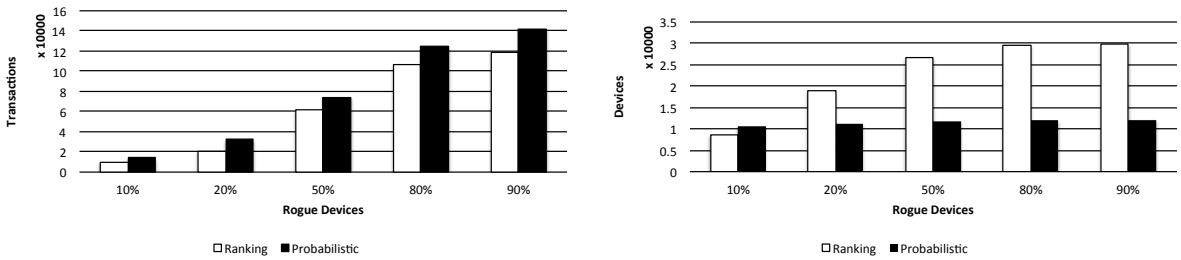
is a particularly challenging environment, given that in a non negligible number of interactions, rogue devices continue to provide useful work for the MC3E.

7.3.5 Reputation Dissemination Using Contact Prediction

The gains of using contact prediction to influence the certificates distributed by the CTE were evaluated using a custom simulator. This simulator was prepared as an alternative to OMNeT++, which lacked the capacities to cope with the data produced by the 2013 MobIPLity dataset.

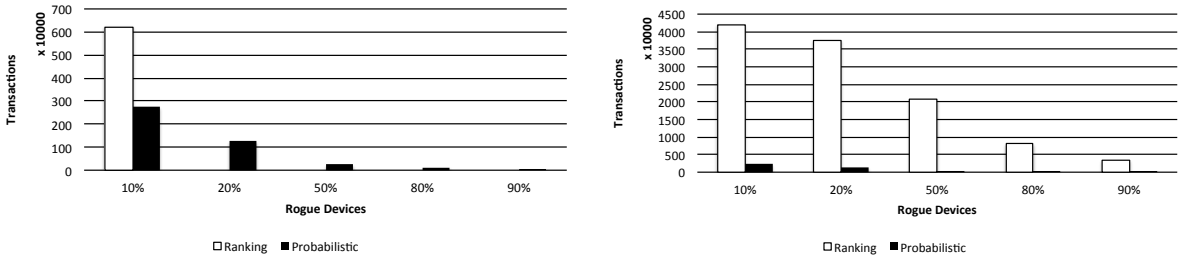
Simulations used a process similar to the one described in Sec. 7.2.2. With MobIPLity and bonnmotion being used to extract all pairs of devices that were within a $50m$ range for at least $60s$. For each contact we considered that a transaction was possible at each $10s$.

Simulation proceeds in days, with all devices connecting to the CTE at midnight. The CTE constructs a ranking of the devices that are expected to be observed on the following day, using the device reports of previous contacts, as described in Sec.7.2. This ranking is then distributed to the devices. On our tests we considered that each device makes available storage space for 100 reputation certificates. The disseminated list includes the top devices that are expected to be observed in the following day and whose reputation is bellow a predefined threshold, we used for this evaluation the



(a) Bad transactions (b) Rogues detected by local cache

Figure 7.10: Improvement of ranking algorithm



(a) Ranking algorithm (without bias) (b) Ranking algorithm biased

Figure 7.11: Successful transactions

value of 0, as seen on Tab. 7.11 which presents the thresholds and other parameters used in the evaluation.

To observe the gains of contact predictions the basic dissemination algorithm (i.e. a biased probabilistic selection of certificates) was equally experimented.

As depicted in Fig. 7.10, an average improvement of 22% and 38%, respectively on the number of bad transactions and detections of rogue devices using local cache of reputation information, is observed when contact prediction is used. The exception is on the number of detected rogues by local cache when the proportion of rogue devices is 10%, which can be attributed to the fact that the probabilistic dissemination algorithm includes any reputation certificates, ordered by the lowest values, while the ranking algorithm used a predefined low threshold to determine which devices have their reputation certificate sent by the CTE. Despite these differences the number of detected bad transactions still shows improvement.

The gains presented by contact prediction motivated to experiment a more ambi-

tious improvement that consisted in the omission of verifying customer peers reputation by inquiring other nodes in the neighbourhood when a certificate is not available in the local cache. The goal would be to reduce the number of messages exchanged by the nodes.

To evaluate this improvement we introduced a bias in the ranking algorithm in order to use 50% of the available local cache at each device to include high ranked devices, according with the contact prediction algorithm, independently of their reputation. The remaining 50% were filled, as before, with the reputation of devices with a threshold bellow 0. Figure 7.11 shows the improvement on the number of good transaction that was able to be obtained when using the biased ranking algorithm when compared with the traditional probabilistic dissemination algorithm. This is further detailed in Fig. 7.12 where the percentage of successful transactions when the biased ranking algorithm is used is depicted according to the source. Possible sources are the local cache or the reputation sent directly by the task peer. The figure also shows the percentage of bad transactions. Between 27% and 34% of all transaction reputation certificates were provided from local cache. However, the use of local reputation will only provide better results at detecting rogue devices. The low number of bad transaction is explained by the benefits of using the contact prediction for the dissemination of reputation certificates, but also because the simulation does not consider pseudonym changes. However, as exposed in Fig. 7.8, their impact would be limited to a delay in the detection of rogues. Little to no impact on the observed improvements are expected by this change, since the CTE includes the current pseudonym when disseminating the reputation certificates.

The impact of this biased ranking algorithm on the previously observed metrics of bad transactions and detected rogue devices by local cache, can be observed in Fig. 7.13, where the improvement is retained.

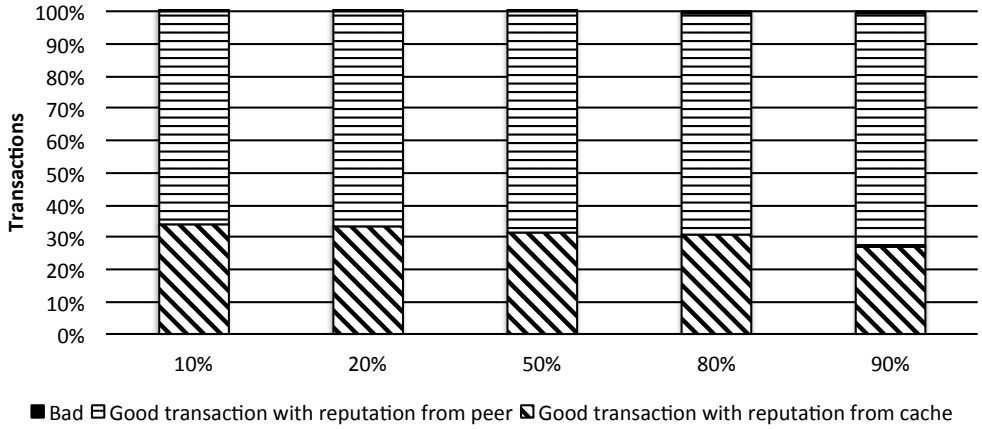
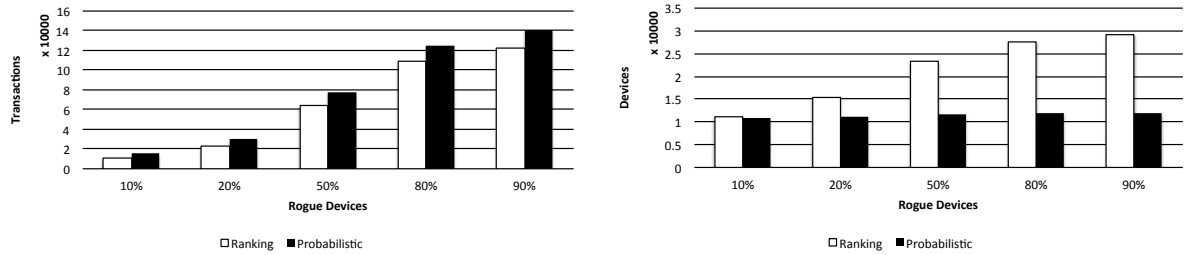


Figure 7.12: Total transactions with ranking algorithm biased



(a) Bad transactions

(b) Rogues detected by local cache

Figure 7.13: Improvement when using a biased ranking

7.4 Summary

This chapter reports on the development of a contact prediction algorithm inspired by contact recurrence of the users observed on the IPL eduroam dataset. Defining a number of different temporal patterns on which recurring contacts occur, the algorithm provides a ranking of the devices that are expected to be in range of any device on any given day. Surprisingly, our findings report that the prediction algorithm was equally applicable on an unrelated dataset, describing the mobility of taxis in the city of Rome.

HTnT, the component of the MC3E responsible for enabling trust and incentive, was experimented with the contact prediction algorithm in order to create a ranking of devices expected to be in range in a near future. The ranking is sorted by the probability of the contact to happen. We have shown that this ranking algorithm outperform a

classic probabilistic dissemination, reducing by as much as 22% the number of transactions with an erroneous outcome.

8

Conclusions and Future Work

Cloud computing is a paradigm where users and companies delegate computation, storage or even management tasks to a third party operator. Cloud services depend of Internet connectivity. This requirement is more noticeable on mobile devices given their mobility onto areas where the Internet connection has lower quality or is unavailable. Our work proposes a way to dismiss this requirement, using mobile devices as nodes of the cloud. We call this architecture the Collaborative Cloudless Computing (C3).

The thesis proposed a framework called Mobile Collaborative Cloudless Computing Environment (MC3E) that addresses the challenges raised by the C3. These challenges appear from the lack of trust between the participants. Privacy of user data, selfish nodes, possibly tampering results, efficient task partitioning and resource search are all examples of problems that need to be resolved before a successful deployment of C3. From these, one of the toughest challenge has its origin on the behaviour of the participants, that demand for a system to enable trust and provide an incentive for users.

To address these problems, the thesis presented the Hybrid Trust and Trade (HTnT) system, that combines reputation with a virtual currency, allowing users to benefit of the C3 proportionally to their contribution, while at the same time providing detection of selfish devices. In addition, the system encourages the use of pseudonyms that can be frequently changed, reducing the use of unique identifiers that could compromise the user privacy. Dissemination of reputation information in HTnT uses a contact pre-

diction algorithm to provide a sub-set of the reputation database, that includes peers predicted to be in range in the near future. In comparison, with a version that uses exclusively the users' reputation for selecting the certificates that will be disseminated, contact prediction improved the detection of rogue devices by as much as 38%. The prediction, facilitates the detection of rogue devices while reducing the burden on the resources of the local device by reducing the number of messages with reputation information between the participants. Using the contact algorithm we were also capable of improving the protocol itself by serving cached reputation certificates for at most 34% of all transactions.

The development of the contact prediction algorithm was inspired by a large dataset of accesses to a number of eduroam network sites at the Polytechnic Institute of Lisbon (IPL) between 2005 and 2013. The dataset was first analysed using a number of generic metrics, such as the variation in the number of users, sessions and traffic with time. Results confirm an increase in the number of users and devices, with the latter growing at a faster pace and that device mobility is changing with users connecting to an increasing number of access points daily. The combination of both results suggests that there is an ongoing change in the use pattern of wireless networks. Users now tend to access to wireless networks through two devices, with one turned on even when the user is moving. However, the increase of the number of devices and connectivity is not reflected in traffic, which tends to decrease (in spite of an increase in mobile devices).

The dataset equally supported the development of MobIPLity, an algorithm that prepares mobility scenarios from observed mobility traces. MobIPLity contrasts with what is commonly found in the literature where scenarios are artificially generated by replicating statistical distributions. A web interface for MobIPLity was publicly made available, allowing developers to rapidly create a scenario in a common, widely adopted format. The availability of mobility scenarios using real world traces contributed to the improvement of the dissemination algorithm of HTnT. However, it can equally be useful for the evaluation of other protocols, applications and algorithms.

The same dataset, was equally used to extract the complete set of temporal com-

munities observed between 2005 and 2013. This set supported the development of a statistical model that characterizes the different temporal community sizes assuming four distinct recurrence patterns that mirror likely schedules of the users. The fitting of the observations showed that the recurrence of three temporal patterns can be modelled by generalized Pareto distributions, confirming the results already observed for the Inter-Contact Times by related work.

The algorithm to predict neighbouring peer devices on a nearby future uses the knowledge of the previous observed contacts and temporal patterns. We evaluate this algorithm using the output of MoIPLity for a different year and an independent trace, describing the movement of taxis in the city of Rome. Results showed that the statistical properties of the temporal communities are shared by both datasets supporting expectations of applicability of the algorithm to distinct environments.

8.1 Future Work

An efficient materialization of the Mobile Collaborative Cloudless Computing Environment (MC3E) cannot be achieved without addressing a number of challenges left open in this thesis. Computing blocks, task delegation, peer search, energy optimization and security and privacy are all components of the MC3E for which no obvious solution can be found.

Concerning the problems related with reputation, future work will be focused on applying HTnT to other environments and further improve the contact prediction algorithm.

The contact prediction algorithm can be improved by considering temporal communities with size larger than two. The evaluation of such a large number of temporal communities has a high computational requirements due to the added complexity of all possible combinations. However, HTnT would benefit of such prediction by decreasing the redundancy in the lists of certificates distributed to each participant. The

prediction can also be improved by considering other metrics or commonly used predictors. Metrics that are expected to improve the algorithm are number of repetitions of a temporal community and their duration.

The dismissal of the central trusted entity (CTE) of the HTnT is also an interesting challenge. As an alternative, it will be challenging investigating mechanisms to hide the real identity of the devices from the CTE.

Experience gathered with the analysis of the dataset can be applied in two directions. One consists in extending it, possibly with data provided by other sources, as the data collection effort at the IPL continues. The other is to investigate novel applications of the data for examples in areas like indoor localization, human mobility prediction and to the evaluation of future network protocols, applications and algorithms.

References

Adams, C. and Lloyd, S. (2002). *Understanding PKI: Concepts, Standards, and Deployment Considerations*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2nd edition.

Anagnostakis, K. G. and Greenwald, M. B. (2004). Exchange-based incentive mechanisms for peer-to-peer file sharing. In *Proceedings of the 24th International Conference on Distributed Computing Systems*, ICDCS 2004, pages 524–533.

Ananthanarayanan, G. and Stoica, I. (2009). Blue-Fi: Enhancing wi-fi performance using Bluetooth signals. In *Proceedings of the 7th International Conference on Mobile Systems, Applications, and Services*, MobiSys '09, pages 249–262, New York, NY, USA. ACM.

Anderson, D. P. (2004). BOINC: a system for public-resource computing and storage. In *Proceedings of the 5th IEEE/ACM International Workshop on Grid Computing*, GRID 2004, pages 4–10.

Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A., Stoica, I., and Zaharia, M. (2010). A view of cloud computing. *Communications of the ACM*, 53(4):50–58.

Aron, J. (2011). How innovative is apple's new voice assistant, siri? *New Scientist*, 212(2836):24.

Aschenbruck, N., Ernst, R., Gerhards-Padilla, E., and Schwamborn, M. (2010). Bonnmotion: A mobility scenario generation and analysis tool. In *Proceedings of the*

3rd International ICST Conference on Simulation Tools and Techniques, SIMUTools '10, pages 51:1–51:10, ICST, Brussels, Belgium, Belgium. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).

Aschenbruck, N., Gerhards-Padilla, E., Gerharz, M., Frank, M., and Martini, P. (2007). Modelling mobility in disaster area scenarios. In *Proceedings of the 10th ACM Symposium on Modeling, Analysis, and Simulation of Wireless and Mobile Systems*, MSWiM '07, pages 4–12, New York, NY, USA. ACM.

Aschenbruck, N., Munjal, A., and Camp, T. (2011). Trace-based mobility modeling for multi-hop wireless networks. *Computer Communications*, 34(6):704–714.

Balani, R. (2007). Energy consumption analysis for bluetooth, wifi and cellular networks. NESL Technical Report TR-UCLA-NESL-200712-01, Networked & Embedded Systems Laboratory.

Bethencourt, J., Sahai, A., and Waters, B. (2007). Ciphertext-policy attribute-based encryption. In *Proceedings of the IEEE Symposium on Security and Privacy*, SP'07, pages 321–334.

Bettstetter, C., Resta, G., and Santi, P. (2003). The node distribution of the random waypoint mobility model for wireless ad hoc networks. *IEEE Transactions on Mobile Computing*, 2(3):257–269.

Bogliolo, A., Polidori, P., Aldini, A., Moreira, W., Mendes, P., Yildiz, M., Ballester, C., and Seigneur, J. (2012). Virtual currency and reputation-based cooperation incentives in user-centric networks. In *Proceedings of the 8th International Wireless Communications and Mobile Computing Conference*, IWCMC 2012, pages 895–900.

Boldrini, C. and Passarella, A. (2010). Hcmm: Modelling spatial and temporal properties of human mobility driven by users' social relationships. *Computer Communications*, 33(9):1056–1074.

Bonomi, F., Milito, R., Zhu, J., and Addepalli, S. (2012). Fog computing and its role

in the internet of things. In *Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing*, MCC '12, pages 13–16, New York, NY, USA. ACM.

Borcea, C., Iyer, D., Kang, P., Saxena, A., and Iftode, L. (2002). Cooperative computing for distributed embedded systems. In *Proceedings of the 22nd International Conference on Distributed Computing Systems*, (ICDCS 2002, pages 227–236.

Bracciale, L., Bonola, M., Loreti, P., Bianchi, G., Amici, R., and Rabuffi, A. (2014). CRAWDAD data set roma/taxi (v. 2014-07-17). Downloaded from http://crawdad.org/roma/taxi/.

Burleigh, S., Hooke, A., Torgerson, L., Fall, K., Cerf, V., Durst, B., Scott, K., and Weiss, H. (2003). Delay-tolerant networking: an approach to interplanetary internet. *IEEE Communications Magazine*, 41(6):128– 136.

Buttyan, L. and Hubaux, J. (2001). Nuglets: a virtual currency to stimulate cooperation in self-organized mobile ad hoc networks. Technical Report DSC/2001/001, Swiss Federal Institute of Technology.

Chafi, H., DeVito, Z., Moors, A., Rompf, T., Sujeeth, A. K., Hanrahan, P., Odersky, M., and Olukotun, K. (2010). Language virtualization for heterogeneous parallel computing. *ACM SIGPLAN Notices - OOPSLA '10*, 45(10):835–847.

Chaintreau, A., Hui, P., Crowcroft, J., Diot, C., Gass, R., and Scott, J. (2007). Impact of human mobility on opportunistic forwarding algorithms. *IEEE Transactions on Mobile Computing*, 6(6):606–620.

Cheng, L. (2002). Service advertisement and discovery in mobile ad hoc networks. In *Proceedings of the ACM Conference on Computer Supported Cooperative Work*, CSCW 2002.

Chun, B.-G., Ihm, S., Maniatis, P., Naik, M., and Patti, A. (2011). Clonecloud: Elastic execution between mobile device and cloud. In *Proceedings of the 6th Conference on Computer Systems*, EuroSys '11, pages 301–314, New York, NY, USA. ACM.

Chun, B. G. and Maniatis, P. (2009). Augmented smartphone applications through clone cloud execution. In *Proceedings of the 12th Workshop on Hot Topics in Operating Systems*, HotOS 2009, page 8.

Clauset, A., Shalizi, C., and Newman, M. (2009). Power-law distributions in empirical data. *SIAM Review*, 51(4):661–703.

Cohen, B. (2003). Incentives build robustness in bittorrent. In *Proceedings of the Workshop on Economics of Peer-to-Peer systems*.

Conti, M., Giordano, S., May, M., and Passarella, A. (2010). From opportunistic networks to opportunistic computing. *IEEE Communications Magazine*, 48(9):126–139.

Costa, P., Mascolo, C., Musolesi, M., and Picco, G. (2008). Socially-aware routing for publish-subscribe in delay-tolerant mobile ad hoc networks. *IEEE Journal on Selected Areas in Communications*, 26(5):748–760.

Cruz, N., Miranda, H., and Ribeiro, P. (2014). The evolution of user mobility on the eduroam network. In *2014 IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*, pages 249–253.

Cuervo, E., Balasubramanian, A., Cho, D.-k., Wolman, A., Saroiu, S., Chandra, R., and Bahl, P. (2010). MAUI: making smartphones last longer with code offload. In *Proceedings of the 8th Annual International Conference on Mobile Systems, Applications and Services (MobiSys 2010)*, MobiSys '10, pages 49–62. ACM. ACM ID: 1814441.

Dean, J. and Ghemawat, S. (2008). MapReduce: simplified data processing on large clusters. *Communications of the ACM*, 51(1):107–113.

Denman, J. (2010). Thoughts on cloud API standards from cloud camp boston 2010. http://itknowledgeexchange.techtarget.com/soa-talk/thoughts-on-cloud-api-standards-from-cloud-camp-boston-2010/ [Last accessed on July 31, 2011].

Dierks, T. (2008). The transport layer security (TLS) protocol version 1.2. Request For Comments 5246, IETF.

Droms, R. (1997). Dynamic Host Configuration Protocol. Technical Report 2131, IETF.

Electro-Mechanics, S. (2010). SWB-B23 datasheet - broadcom BCM4329 WLAN+BT solution.

Eugster, P. T., Felber, P. A., Guerraoui, R., and Kermarrec, A. M. (2003). The many faces of publish/subscribe. *ACM Computing Surveys (CSUR)*, 35(2):114–131.

Feeney, L. (2001). An energy consumption model for performance analysis of routing protocols for mobile ad hoc networks. *Mobile Networks and Applications*, 6(3):239–249.

Felix Marmol, J. G. and Perez, G. (2010). Trims, a privacy-aware trust and reputation model for identity management systems. *Computer Networks*, 54:2899–2912.

Fernandes, A., Kotsovinos, E., Östring, S., and Dragovic, B. (2004). Pinocchio: Incentives for honest participation in distributed trust management. In Jensen, C., Poslad, S., and Dimitrakos, T., editors, *Trust Management*, volume 2995 of *Lecture Notes in Computer Science*, pages 63–77. Springer Berlin Heidelberg.

Fernando, N., Loke, S. W., and Rahayu, W. (2013). Mobile cloud computing: A survey. *Future Generation Computer Systems*, 29(1):84–106. Including Special section: AIRCC-NetCoM 2009 and Special section: Clouds and Service-Oriented Architectures.

Ferreira, D., Dey, A., and Kostakos, V. (2011). Understanding human-smartphone concerns: A study of battery life. In Lyons, K., Hightower, J., and Huang, E., editors, *Pervasive Computing*, volume 6696 of *Lecture Notes in Computer Science*, pages 19–33. Springer Berlin Heidelberg.

Fischer, D., Herrmann, K., and Rothermel, K. (2010). Gesomo: A general social mobility model for delay tolerant networks. In *Proceedings of the IEEE 7th International Conference on Mobile Adhoc and Sensor Systems*, MASS 2010, pages 99–108.

Foell, S., Phithakkitnukoon, S., Kortuem, G., Veloso, M., and Bento, C. (2014). Catch me if you can: Predicting mobility patterns of public transport users. In *Proceedings*

of the IEEE 17th International Conference on Intelligent Transportation Systems, ITSC 2014, pages 1995–2002.

Garland, M., Grand, S. L., Nickolls, J., Anderson, J., Hardwick, J., Morton, S., Phillips, E., Zhang, Y., and Volkov, V. (2008). Parallel computing experiences with CUDA. *IEEE Micro*, 28(4):13–27.

Ghemawat, S., Gobioff, H., and Leung, S. (2003). The google file system. In *Proceedings of the 9th ACM Symposium on Operating Systems Principles*, volume 37 of *SOSP 2003*, pages 29–43.

Gonzalez, M. C., Hidalgo, C. A., and Barabasi, A.-L. (2008). Understanding individual human mobility patterns. *Nature*, 453(7196):779–782.

Google (2015). Art and dalvik. `http://source.android.com/devices/tech/dalvik/`. [Online; accessed 11-March-2015].

Gyarmati, M., Schilcher, U., Brandner, G., Bettstetter, C., Chung, Y. W., and Kim, Y.-H. (2008). Impact of random mobility on the inhomogeneity of spatial distributions. In *Proceedings of the IEEE Global Telecommunications Conference*, GLOBECOM 2008, pages 1–5.

Henderson, T., Kotz, D., and Abyzov, I. (2008). The changing usage of a mature campus-wide wireless network. *Computer Networks*, 52(14):2690–2712.

Hendrikx, F., Bubendorfer, K., and Chard, R. (2015). Reputation systems: A survey and taxonomy. *Journal of Parallel and Distributed Computing*, 75(0):184–197.

Hill, J., Szewczyk, R., Woo, A., Hollar, S., Culler, D., and Pister, K. (2000). System architecture directions for networked sensors. *ACM SIGPLAN Notices*, 35(11):93–104.

Hoque, M., Siekkinen, M., and Nurminen, J. (2011). On the energy efficiency of proxy-based traffic shaping for mobile audio streaming. In *Proceedings of the IEEE Consumer Communications and Networking Conference*, CCNC 2011), pages 891–895.

Houser, D. and Wooders, J. (2006). Reputation in auctions: Theory, and evidence from ebay. *Journal of Economics and Management Strategy*, 15:353–369.

Hu, J. and Burmester, M. (2009). Cooperation in mobile ad hoc networks. In Misra, S., Woungang, I., and Chandra Misra, S., editors, *Guide to Wireless Ad Hoc Networks*, Computer Communications and Networks, pages 43–57. Springer London.

Huang, D., Zhang, S., Hui, P., and Chen, Z. (2015). Link pattern prediction in opportunistic networks with kernel regression. In *Proceedings of The 7th International Conference on COMmunication Systems and NETworkS*, COMSNETS 2015.

Huang, D., Zhang, X., Kang, M., and Luo, J. (2010). MobiCloud: building secure cloud framework for mobile computing and communication. In *Proceedings of the 5th IEEE International Symposium on Service Oriented System Engineering*, SOSE 2010, pages 27–34.

Huerta-Canepa, G. and Lee, D. (2010). A virtual cloud computing provider for mobile devices. In *Proceedings of the 1st ACM Workshop on Mobile Cloud Computing & Services: Social Networks and Beyond*, MCS 2010, pages 1–5.

Hui, P., Crowcroft, J., and Yoneki, E. (2011). Bubble rap: Social-based forwarding in delay-tolerant networks. *IEEE Transactions on Mobile Computing*, 10(11):1576–1589.

Iosifidis, G., Gao, L., Huang, J., and Tassiulas, L. (2014). Enabling crowd-sourced mobile internet access. In *Proceedings of the 33rd IEEE International Conference on Computer Communications*, INFOCOM 2014, pages 451–459.

Johnson, D. B. and Maltz, D. A. (1996). Dynamic source routing in ad hoc wireless networks. In Imielinski, T. and Korth, H., editors, *Mobile Computing*, pages 153–181. Kluwer Academic Publishers.

Johnson, D. B., Maltz, D. A., Broch, J., et al. (2001). DSR: the dynamic source routing protocol for multi-hop wireless ad hoc networks. *Ad hoc networking*, 5:139–172.

Jun, S. and Ahamad, M. (2005). Incentives in BitTorrent induce free riding. In *Proceedings of the 3rd ACM SIGCOMM Workshop on Economics of Peer-to-Peer Systems*, P2PECON 2005, pages 116–121.

Kamvar, S. D., Schlosser, M. T., and Garcia-Molina, H. (2003). The eigentrust algorithm for reputation management in P2P networks. In *Proceedings of the 12th international conference on World Wide Web*, WWW '03, pages 640–651, New York, NY, USA. ACM.

Kangasharju, J., Ott, J., and Karkulahti, O. (2010). Floating content: Information availability in urban environments. In *Proceedings of the 8th IEEE International Conference on Pervasive Computing and Communications Workshops*, PERCOM 2010, pages 804–808.

Karagiannis, T., Le Boudec, J.-Y., and Vojnovic, M. (2010). Power law and exponential decay of intercontact times between mobile devices. *Mobile Computing, IEEE Transactions on*, 9(10):1377–1390.

Karamshuk, D., Boldrini, C., Conti, M., and Passarella, A. (2011). Human mobility models for opportunistic networks. *IEEE Communications Magazine*, 49(12):157–165.

Karlson, A., Meyers, B., Jacobs, A., Johns, P., and Kane, S. (2009). Working overtime: Patterns of smartphone and pc usage in the day of an information worker. In Tokuda, H., Beigl, M., Friday, A., Brush, A., and Tobe, Y., editors, *Pervasive Computing*, volume 5538 of *Lecture Notes in Computer Science*, pages 398–405. Springer Berlin Heidelberg.

Kim, M., Kotz, D., and Kim, S. (2006). Extracting a mobility model from real user traces. In *Proceedings of the 25th IEEE International Conference on Computer Communications*, INFOCOM 2006, pages 1–13.

Kumar, K. and Lu, Y. H. (2010). Cloud computing for mobile users: can offloading computation save energy? *Computer*, 43(4):51–56.

Lee, K., Hong, S., Kim, S. J., Rhee, I., and Chong, S. (2009). Slaw: A new mobility model for human walks. In *Proceedings of the 29th IEEE International Conference on Computer Communications*, INFOCOM 2009, pages 855–863.

Lindgren, A., Doria, A., and Schelén, O. (2003). Probabilistic routing in intermittently connected networks. *ACM SIGMOBILE Mobile Computing and Communications Review*, 7(3):19–20.

Luttenberger, N. and Peters, H. (2011). A resilient and energy-saving incentive system for resource sharing in MANETs. In *Proceedings of the 17th GI/ITG Conference on Communication in Distributed Systems*, volume 17 of *KiVS 2011*, pages 109–120.

Madden, S., Franklin, M. J., Hellerstein, J. M., and Hong, W. (2002). Tag: a tiny aggregation service for ad-hoc sensor networks. *ACM SIGOPS Operating Systems Review*, 36(SI):131–146.

Mahmoud, M. and Shen, X. (2011). Esip: Secure incentive protocol with limited use of public-key cryptography for multihop wireless networks. *IEEE Transactions on Mobile Computing*, 10(7):997–1010.

Mahmoud, M. E. and Shen, X. (2010). Stimulating cooperation in multi-hop wireless networks using cheating detection system. In *Proceedings of the 29th IEEE International Conference on Computer Communications*, INFOCOM 2010, pages 1–9.

Marinelli, E. E. (2009). *Hyrax: Cloud Computing on Mobile Devices using MapReduce*. PhD thesis, Carnegie-Mellon University.

Maymounkov, P. and Mazières, D. (2002). Kademlia: A peer-to-peer information system based on the XOR metric. In Druschel, P., Kaashoek, F., and Rowstron, A., editors, *Peer-to-Peer Systems*, volume 2429 of *Lecture Notes in Computer Science*, pages 53–65. Springer Berlin Heidelberg.

McCloghrie, K., McCloghrie, K., Schoenwaelder, J., and Perkins, D. (1999). Structure of management information version 2 (SMIv2). Request For Comments 2578, IETF.

Mell, P. and Grance, T. (2009). Effectively and securely using the cloud computing paradigm. *NIST, Information Technology Laboratory*.

Meshkova, E., Riihijarvi, J., Petrova, M., and Mahonen, P. (2008). A survey on resource discovery mechanisms, peer-to-peer and service discovery frameworks. *Computer networks*, 52(11):2097–2128.

Metsch, T. (2010). Open cloud computing interface: Use cases and requirements for a cloud API. *Open Grid Forum, GDF-I*, 162.

Miranda, H., Leggio, S., Rodrigues, L., and Raatikainen, K. (2007). An algorithm for dissemination and retrieval of information in wireless ad hoc networks. In Kermarrec, A.-M., Bougé, L., and Priol, T., editors, *Euro-Par 2007 Parallel Processing*, volume 4641 of *Lecture Notes in Computer Science*, pages 891–900. Springer Berlin Heidelberg.

Miranda, H. and Rodrigues, L. (2006). A framework to provide anonymity in reputation systems. In *Proceedings of the 3rd Annual International Conference on Mobile and Ubiquitous Systems: Networking Services*, MobiQuitous 2006, pages 1–4.

Miranda, H. and Rodrigues, L. (2010). Reputation in anonymous vehicular networks. *Journal of Autonomous and Adaptive Communications Systems*, 3(2):178–197.

Mulhanga, M., Lima, S., and Carvalho, P. (2011). Characterising university WLANs within eduroam context. In Balandin, S., Koucheryavy, Y., and Hu, H., editors, *Smart Spaces and Next Generation Wired/Wireless Networking*, volume 6869 of *Lecture Notes in Computer Science*, pages 382–394. Springer Berlin Heidelberg.

Murray, D. G., Yoneki, E., Crowcroft, J., and Hand, S. (2010). The case for crowd computing. In *Proceedings of the 2nd ACM SIGCOMM Workshop on Networking, Systems, and Applications on Mobile Handhelds*, MobiHeld 2010, pages 39–44.

Nakamoto, S. (2008). Bitcoin: A Peer-to-Peer Electronic Cash System.

Nash, J. (1950). The bargaining problem. *Econometrica: Journal of the Econometric Society*, 18(2):155–162.

Neumann, J. V. and Morgenstern, O. (1944). *Theory of Games and Economic Behavior*. Princeton University Press.

Newman, M. E. J. (2004). Analysis of weighted networks. *Physical Review E*, 70:056131.

Orlinski, M. and Filer, N. (2013). The rise and fall of spatio-temporal clusters in mobile ad hoc networks. *Ad Hoc Networks*, 11(5):1641–1654.

Palla, G., Derenyi, I., Farkas, I., and Vicsek, T. (2005). Uncovering the overlapping community structure of complex networks in nature and society. *Nature*, 435(7043):814–818.

Pelusi, L., Passarella, A., and Conti, M. (2006). Opportunistic networking: data forwarding in disconnected mobile ad hoc networks. *IEEE Communications Magazine*, 44(11):134–141.

Pering, T., Agarwal, Y., Gupta, R., and Want, R. (2006). CoolSpots: reducing the power consumption of wireless mobile devices with multiple radio interfaces. In *Proceedings of the 4th International Conference on Mobile Systems, Applications, and Services*, MobiSys 2006, pages 220–232.

Perkins, C., Belding-Royer, E., and Das, S. (2003). Ad hoc on-demand distance vector (AODV) routing. Request For Comments 3561, IETF.

Perkins, C. and Wang, K.-Y. (1999). Optimized smooth handoffs in mobile ip. In *Proceedings of the IEEE International Symposium on Computers and Communications*, pages 340–346.

Perrucci, G., Fitzek, F., Sasso, G., Kellerer, W., and Widmer, J. (2009). On the impact of 2g and 3g network usage for mobile phones' battery life. In *Proceedings of the 15th European Wireless Conference*, EW 2009, pages 255–259.

Pietiläinen, A.-K. and Diot, C. (2012). Dissemination in opportunistic social networks: The role of temporal communities. In *Proceedings of the 13th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, MobiHoc '12, pages 165–174, New York, NY, USA. ACM.

Piorkowski, M., Sarafijanovic-Djukic, N., and Grossglauser, M. (2009). CRAWDAD data set epfl/mobility (v. 2009-02-24). Downloaded from http://crawdad.cs.dartmouth.edu/epfl/mobility.

Pitkänen, M., Kärkkäinen, T., Ott, J., Conti, M., Passarella, A., Giordano, S., Puccinelli, D., Legendre, F., Trifunovic, S., Hummel, K., May, M., Hegde, N., and Spyropoulos, T. (2012). SCAMPI: Service platform for social aware mobile and pervasive computing. *ACM SIGCOMM Computer Communication Review*, 42(4):503–508.

Ratnasamy, S., Francis, P., Handley, M., Karp, R., and Shenker, S. (2001). A scalable content-addressable network. *SIGCOMM Computer Communication Review*, 31(4):161–172.

Resnick, P., Kuwabara, K., Zeckhauser, R., and Friedman, E. (2000). Reputation systems. *Communications of the ACM*, 43(12):45–48.

Rigney, C. (2000). RADIUS Accounting. Technical Report 2866, IETF.

Rowstron, A. and Druschel, P. (2001). Pastry: Scalable, decentralized object location, and routing for Large-Scale Peer-to-Peer systems. In Guerraoui, R., editor, *Middleware 2001*, volume 2218 of *Lecture Notes in Computer Science*, pages 329–350. Springer Berlin / Heidelberg.

Satyanarayanan, M. (2001). Pervasive computing: vision and challenges. *Personal Communications, IEEE*, 8(4):10–17.

Satyanarayanan, M., Bahl, P., Caceres, R., and Davies, N. (2009). The case for vm-based cloudlets in mobile computing. *Pervasive Computing, IEEE*, 8(4):14–23.

Satyanarayanan, M., Schuster, R., Ebling, M., Fettweis, G., Flinck, H., Joshi, K., and Sabnani, K. (2015). An open ecosystem for mobile-cloud convergence. *IEEE Communications Magazine*, 53(3):63–70.

Schalkwyk, J., Beeferman, D., Beaufays, F., Byrne, B., Chelba, C., Cohen, M., Kamvar, M., and Strope, B. (2010). *Google Search by Voice: A case study*. Springer.

Schilcher, U., Gyarmati, M., Bettstetter, C., Chung, Y. W., and Kim, Y.-H. (2008). Measuring inhomogeneity in spatial distributions. In *Proceedings of the IEEE Vehicular Technology Conference*, VTC 2008, pages 2690–2694.

Schneier, B. (1995). *Applied Cryptography: Protocols, Algorithms, and Source Code in C*. John Wiley & Sons, Inc., New York, NY, USA, 2nd edition.

Singh, A. and Liu, L. (2003). Trustme: anonymous management of trust relationships in decentralized P2P systems. In *Proceedings of the 3rd International Conference on P2P Computing*, P2P 2003.

Song, C., Koren, T., Wang, P., and Barabási, A.-L. (2010a). Modelling the scaling properties of human mobility. *Nature Physics*, 6(10):818–823.

Song, C., Qu, Z., Blumm, N., and Barabási, A.-L. (2010b). Limits of predictability in human mobility. *Science*, 327(5968):1018–1021.

Stallings, W. (2010). *Cryptography and network security: principles and practice*. Prentice Hall Press, 5th edition.

Stoica, I., Morris, R., Karger, D., Kaashoek, M. F., and Balakrishnan, H. (2001). Chord: A scalable peer-to-peer lookup service for internet applications. *ACM SIGCOMM Computer Communication Review*, 31(4):149–160.

Su, J., Scott, J., Hui, P., Crowcroft, J., de Lara, E., Diot, C., Goel, A., Lim, M., and Upton, E. (2007). Haggle: Seamless networking for mobile applications. In Krumm, J., Abowd, G., Seneviratne, A., and Strang, T., editors, *UbiComp 2007: Ubiquitous Computing*, volume 4717 of *Lecture Notes in Computer Science*, pages 391–408. Springer Berlin Heidelberg.

Tan, W. L., Lam, F., and Lau, W. C. (2008). An empirical study on the capacity and performance of 3G networks. *IEEE Transactions on Mobile Computing*, 7(6):737–750.

Tang, D. and Baker, M. (2000). Analysis of a local-area wireless network. In *Proceedings of the 6th Annual International Conference on Mobile computing and networking*, Mobicom'00, pages 1–10. ACM.

Thiagarajan, A., Ravindranath, L., Balakrishnan, H., Madden, S., and Girod, L. (2011). Accurate, low-energy trajectory mapping for mobile devices. In *Proceedings of the 8th USENIX Conference on Networked Systems Design and Implementation*, NSDI'11, pages 20–20, Berkeley, CA, USA. USENIX Association.

Thies, W., Karczmarek, M., and Amarasinghe, S. (2002). StreamIt: a language for streaming applications. In Horspool, R. N., editor, *Compiler Construction*, volume 2304, pages 179–196. Springer Berlin Heidelberg.

White, T. (2009). *Hadoop: The Definitive Guide*. O'Reilly Media, Inc.

Wortham, J. (2009). Customers angered as iphones overload 3g. http://www.nytimes.com/2009/09/03/technology/companies/03att.html,2009. [Last accessed on September 28, 2011].

Xing, T., Huang, D., Ata, S., and Medhi, D. (2013). Mobicloud: A geo-distributed mobile cloud computing platform. In *Proceedings of the 8th International Conference on Network and Service Management*, CNSM '12, pages 164–168, Laxenburg, Austria, Austria. International Federation for Information Processing.